UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

The development of a test facility for satellite experiments

Author: Daniel Christopher Jansen
Supervisor: Dr Lourens Visagie
Co-supervisor: Dr Willem Jordaan

Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering at the Faculty of Engineering, Stellenbosch University

March 2021

Parts of the work done in this thesis have been published as an international conference article.

Jansen, D.C, Visagie, L., Jordaan, H.W. 2020. An Experimental Investigation Of 3D Printed Cold Gas Thruster Nozzles. *2020 International SAUPEC/RobMech/PRASA Conference*. Cape Town, South Africa.

Plagiarism declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

March 2021

Date

Signature

Abstract

This thesis addresses the problem of designing and implementing a planar air bearing test facility for satellite experiments. The test facility's purpose is to provide a testbed for researchers to validate their projects and to perform experiments related to the field of spacecraft and space systems. The scope of the project includes the design and implementation of an air bearing vehicle, a pose estimator which measures the states of the vehicles being operated on the test facility, and a demonstration payload which can perform a control manoeuvre with the vehicle to showcase the test facility's capabilities.

An air bearing vehicle was designed and constructed, which successfully facilitated low friction movement on a glass platform. This low friction movement emulates the orbital dynamics of satellites with three degrees of freedom. A supporting propellant refill station was also designed and constructed to enable the refilling of the vehicle's gas supply between experiments. The air bearing vehicle carries a payload which typically consists of an on-board computer (which can control the vehicle), along with any other hardware which is specific to the experiment being conducted on the test facility.

A custom cold gas thruster nozzle was designed for use on the air bearing vehicle. This nozzle was manufactured using a fused deposition modelling 3D printer. The nozzles used on the vehicle were characterised using a custom-built test rig and were found to meet all specifications required for the vehicle.

A feedback and communication system (FCS) was designed and implemented, which is responsible for measuring the positions and velocities of the vehicles on the table. This data is logged and displayed to the user in near-real time. The FCS can also wirelessly communicate pose information to the payloads carried by the vehicles and can receive relevant info from the payloads to display and log.

A demonstration payload was designed to showcase the capabilities and functioning of the test facility. This payload includes a control system that controls the vehicle to perform a manoeuvre to move from an initial coordinate to a target coordinate. The demonstration payload successfully showcased the test facility.

Aside from some minor issues, the practical implementation of the test facility performed well and successfully met all requirements. The test facility will provide many advantages for future researchers, allowing them to validate and demonstrate their projects, as well as give them valuable practical experience.

Uittreksel

Hierdie proefskrif behandel die probleem van die ontwerp en implementering van 'n lugdraende toetsfasiliteit vir satellieteksperimente. Die doel van die toetsfasiliteit is om 'n toetsbed aan navorsers te bied om hul projekte te bekragtig en om eksperimente uit te voer wat verband hou met die veld van ruimtetuie en ruimtestelsels. Die omvang van die projek sluit in die ontwerp en implementering van 'n lugdraende voertuig, 'n orientasie-afskatter wat die toestande van die voertuie wat op die toetsfasiliteit gebruik word, meet, en 'n demonstrasie-loonvrag wat 'n beheer van die voertuig kan uitvoer om die funksies van die fasiliteit te vertoon.

'n Lugdraende voertuig is ontwerp en gebou, wat lae wrywing op 'n glasplatform suksesvol moontlik gemaak het. Hierdie beweging met lae wrywing boots die wenteldinamika van satelliete met drie vryheidsgrade na. 'n Ondersteunende dryfstofvulstasie is ook ontwerp en gebou om die hervul van die gastoevoer tussen eksperimente moontlik te maak. Die lugdraende voertuig dra 'n loonvrag wat gewoonlik bestaan uit 'n aanboordrekenaar (wat die voertuig kan beheer), saam met enige ander hardeware wat spesifiek is vir die eksperiment wat op die toetsfasiliteit gedoen word.

'n Spesiale spuitkop vir 'n stuwer is ontwerp vir gebruik op die lugdraende voertuig. Hierdie spuitkop is vervaardig met behulp van 'n versmelte neerslagmodelle 3D-drukker. Die spuitpunte wat op die voertuig gebruik word, is gekenmerk deur 'n pasgemaakte toetsbuis en daar was gevind dat dit aan alle spesifikasies voldoen.

'n Terugvoer- en kommunikasiestelsel is ontwerp en geïmplementeer wat verantwoordelik is vir die meting van die posisies en snelhede van die voertuie op die tafel. Hierdie data word aangeteken en in die regte tyd aan die gebruiker vertoon. Die Terugvoer- en kommunikasiestelsel kan ook draadlose posisie-inligting kommunikeer aan die loonvragte wat deur die voertuie vervoer word en kan relevante inligting van die loonvragte ontvang om te vertoon en aan te meld.

'n Demonstrasie-loonvrag is ontwerp om die funksies en funksionering van die toetsfasiliteit ten toon te stel. Hierdie loonvrag bevat 'n beheerstelsel wat die voertuig beheer om van 'n aanvanklike koördinaat na 'n teikenkoördinaat te beweeg. Die demonstrasie-loonvrag het die toetsfasiliteit suksesvol ten toon gestel.

Afgesien van 'n paar klein probleme, werk die praktiese implementering van die toetsfasiliteit goed en voldoen dit aan alle vereistes. Die toetsfasiliteit bied baie voordele vir toekomstige navorsers, wat hulle in staat stel om hul projekte te bekragtig en te demonstreer, sowel as waardevolle praktiese ervaring te verkry.

Acknowledgements

I would like to extend my thanks to the following:

- To my Heavenly Father, for His unending love and for providing strength.
- To Dr Lourens Visagie and Dr Willem Jordaan, for their guidance, enthusiasm, advice, and support throughout the project.
- To my family, Danie Jansen, Goretti Jansen and Chandré Jansen, and my girlfriend Nicole Taylor for all the love, support, and motivation.
- To my fellow ESL students for creating a great work environment in the lab and to my friends in the ESL, Henry Kotzé, Armand Scholtz, Ruan Viljoen and Robert Waller for the input, the motivation, and the banter.
- To the staff at the Electrical & Electronic Engineering Faculty of Stellenbosch University, including Jenny Martin, Wessel Croukamp and Gift Lecholo for their assistance with procurement and manufacturing.

Table of contents

List of figures

List of tables

List of acronyms and abbreviations

ADC – Analog-to-digital converter

CAD – Computer-aided design

CGT – Cold gas thruster

CoM – centre of mass

DC – Direct current

ESL – Electronic Systems Laboratory

FCS – Feedback and communication system

FDM – Fused deposition modelling

GND – Ground

GUI – Graphical user interface

HIL – Hardware-in-loop

LED – Light emitting diode

MOSFET – Metal-oxide-semiconductor field-effect transistor

NTC – Negative temperature coefficient

PLA – Polylactic acid

PWM – Pulse-width modulation

ROS – Robot Operating System

RW – Reaction wheel

SDK – Software development kit

SISO – Single-input single-output

UART – Universal asynchronous receiver/transmitter

USB – Universal serial bus

1 Introduction

1.1 Background

The expensive and risky nature of the design and manufacture of spacecraft components creates a need to thoroughly test and validate hardware and software before they are used on real missions. This is where satellite test facilities (also known as spacecraft simulators) come in. These facilities are designed to partially emulate the conditions that the spacecraft will face in orbit, most often focusing on emulating the dynamics of the spacecraft in micro-gravity environments. There are a few common variants of satellite test facilities – those which focus on spacecraft dynamics include suspension systems, neutral buoyancy systems, and air bearing test facilities (Wilde et al., 2019).

The Electronic Systems Laboratory (ESL) research group at Stellenbosch University focuses on the control of vehicles and robotics, including spacecraft and space systems. A satellite test facility is desired for ESL researchers to test and validate their spacecraft components on, to serve as a testbed to give researchers practical experience, and to provide a platform to demonstrate their research projects. This project aims to develop a test facility for this purpose. Currently, the ESL has an archaic air bearing vehicle and a large glass table atop which this vehicle hovered during experiments. The previous air bearing vehicle and the glass table are photographed in Figure 1.1. An air bearing is a device which creates a pressurised film of gas between itself and the surface which it hovers on, establishing an interface with very low friction (Schwartz et al., 2003). This is used to emulate the low-friction environment that spacecraft face in orbit. Since the glass table is already available, this project will focus on developing a planar air bearing test facility for satellite experiments.

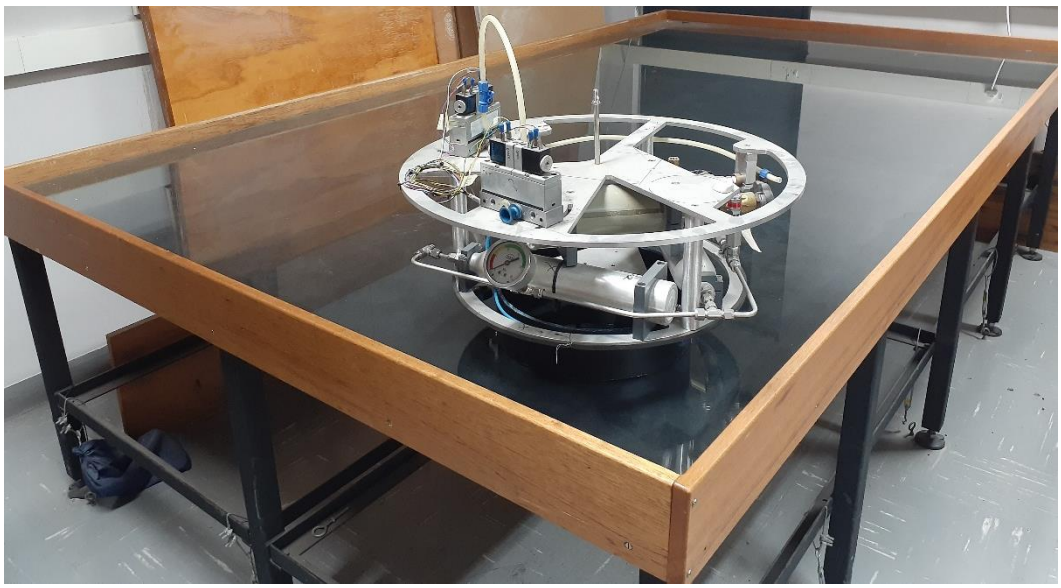


Figure 1.1 - A photograph of the outdated air bearing vehicle and the glass platform previously used for low-friction experiments in the ESL.

1.2 Motivation

Having access to an air bearing test facility yields many benefits for researchers working in space-related fields. It will provide researchers with a testbed which can augment their research in various ways.

The test facility will allow researchers to test and validate their projects in a practical environment, and to perform experiments required for their research. Types of experiments that can be performed on air bearing test facilities include testing of fixed-base and free-floating manipulators, formation flying, docking interactions, debris capture, tether experiments, and more (Rybus & Seweryn, 2016).

The test facility can also act as a testbed for general robotics experiments and does not need to be constrained to spacecraft applications. This opens the facility up for use by ESL researchers in other fields of research. For example, small wheeled robots can be driven on the table and the pose estimator can be used for measuring their states. This setup could be used for testing projects related to problems such as collision avoidance and path planning.

Using the test facility, future researchers can gain experience with practical issues related to their projects. Practical tests provide valuable insights which simulations simply cannot provide. This is an invaluable source of engineering experience, which will potentially improve the future projects of the researchers and the ESL.

Finally, the test facility can be used as a demonstration platform to showcase the projects of the ESL to interested parties. For example, the test facility can be used on Stellenbosch University's Engineering Open Days to exhibit working projects to attract prospective students. Additionally, working projects or concepts can be demonstrated to industry partners.

1.3 Project scope

This thesis entails the design, construction, and practical implementation of a planar air bearing satellite test facility. The scope of the project includes the design of an air bearing vehicle, as well as a pose estimator to measure the positions and velocities of multiple vehicles on the glass table. The air bearing vehicle should support and transport a payload which will generally consist of an on-board computer and auxiliary hardware that future users want to validate on the test facility. This payload will be able to control the motion of the air bearing vehicle. A demonstration payload with a basic control system will be designed and implemented to demonstrate the capabilities and functionality of the test facility in practice. The test facility should be safe for researchers to use, should be expandable and versatile to allow for various types of experiments to take place, and should be easy to use and maintain.

The planar air bearing test facility should be capable of facilitating experiments such as:

- Rendezvous and docking.
- Formation flying.
- Free-floating manipulators.
- Debris capture.
- Tethered spacecraft.
- Solar panel, antenna, or sail deployments.
- General low friction motion tests.

1.4 Methodology

The methodological steps followed to solve the problem defined in the project scope are presented in this section.

Firstly, a literature study will be performed to determine the trends in the design and implementation of satellite test facilities. This knowledge will be applied to the development of the test facility.

Following the literature study, a system-level design of the test facility will be performed. The requirements of the test facility will be determined based on the needs of the ESL.

After the test facility's system architecture is defined, the design and construction of the air bearing vehicle will be focused on. Along with this, any supporting systems such as a gas refill station will be designed and constructed. Once the air bearing vehicle is ready for implementation, some basic tests will be done to confirm that the design functions as expected.

Ensuing the completion of the air bearing vehicle's construction, the pose estimator and any required supporting systems will be designed and implemented. When this is complete, some basic tests will be done to confirm that the design operates as expected.

Subsequent to this, a demonstration payload will be designed and implemented. This payload will include a control system which can showcase the functionality of the test facility by performing a basic manoeuvre using the air bearing vehicle.

Finally, the test facility's practical implementation will be evaluated and demonstrated. This will involve measuring of specifications, identification of practical issues and possible solutions, and finally recording the results from a manoeuvre performed by the demonstration payload.

1.5 Thesis outline

The content outline of this thesis is presented in this section.

Chapter 2 summarises the literature study performed. The study covers satellite test facilities in general, with a more detailed look at planar air bearing test facilities.

Chapter 3 details the system-level design of the test facility. It begins by outlining the test facility's design requirements. Following this, the system architecture and a system-level functional description of the test facility are discussed. Finally, an overview is given of the functional description and design requirements of each subsystem.

Chapter 4 covers the design of the hardware elements of the test facility. Emphasis is placed on the design of the air bearing vehicle (named the carrier cart), but the gas refill station and demonstration payload's hardware are also discussed.

Chapter 5 details the design and characterisation of custom cold gas thruster nozzles used on the air bearing vehicle. The nozzles are designed to be manufactured using a fused deposition modelling 3D printer.

Chapter 6 discusses the design of the feedback and communication system (FCS), which is a subsystem of the test facility. The FCS includes the optical pose estimator, a wireless communication system capable of providing feedback, and a user interface which allows users to view and log relevant data.

Chapter 7 looks at the design of the demonstration payload's control system. This includes the modelling of the carrier cart's dynamics, a pure simulation of the control system and a hardware-in-loop simulation of the control system.

Chapter 8 discusses the practical implementation of the test facility and the results related to the system's performance and functioning. The discussion includes the performance of the carrier cart and the FCS, and highlights problems faced by the practical implementation of the test facility. Furthermore, the results obtained from a control manoeuvre performed by the demonstration payload are presented.

Chapter 9 concludes the thesis, providing an overview of the work done. Additionally, some recommendations are made for the use and expansion of the test facility.

2 Literature study

This chapter discusses the various literature relevant to the design and implementation of planar satellite test facilities based on air bearings. Firstly, an overview is given of the history of air bearing test facilities and their applications. Following this, the typical features and components seen in existing planar air bearing test facilities are discussed.

2.1 An overview of air bearing satellite test facilities

2.1.1 An overview of general satellite test facilities

The nature of designing hardware and software for space systems and the strict requirements thereof cause these components to have high costs and high risks associated with them. It is therefore necessary to rigorously test and verify the functioning of these components before using them in real applications. This is where the need for various types of satellite test facilities is introduced. The purpose of a satellite test facility is to partially mimic orbital conditions, the most notable of which is the condition of micro-gravity. There are numerous types of test facilities (also known as spacecraft simulators) which emulate micro-gravity conditions, as outlined by Menon et. al (2007) and Wilde et al. (2019). Aside from air bearing test facilities, other types include:

- Suspension systems – which are rather complex mechanical systems that compensate for forces applied to the satellite. In active configurations the satellite is typically suspended using cables attached to either motors or mechanical arms, and micro-gravity environments with as low as 0.01g can be achieved. The main problem with these types of test facilities is that they tend to induce significant disturbances on the hardware tested. (Menon et. al, 2007; Wilde et al., 2019).
- Parabolic flight test facilities – which makes use of an aircraft to momentarily achieve micro-gravity environments in which satellite components can be tested. With these test facilities it is possible to achieve 0g, but practically there are some variations on the order of 0.05g in the normal accelerations during the micro-gravity phase of the flight. The main issue faced with this kind of test facility is the size restrictions and short experiment durations, which are typically around 20 s long. (Menon et. al, 2007; Wilde et al., 2019).
- Underwater test facilities – which requires the design of a neutrally buoyant vessel to house the hardware to be tested. Special care needs to be taken to protect components from water damage. The propulsion systems designed for these vessels need to minimise disturbances on the surrounding underwater environment and fluid drag needs to be accounted for. The key disadvantage of this kind of test facility is the need to design a custom vessel

for each payload. Furthermore, these kinds of facilities see more use in training astronauts for spaceflight. (Menon et. al, 2007; Wilde et al., 2019).

This project focuses on building a planar air bearing test facility, since a glass platform is readily available in the laboratory. Air bearing test facilities are some of the most common types constructed by research groups. These facilities emulate micro-gravity conditions by creating a low-friction interface between the satellite and the planar platform upon which they move. (Menon et. al, 2007).

2.1.2 History and overview of air bearing test facilities

Schwartz et al. (2003) have conducted a study of the history of air bearing test facilities. The use of air bearings for spacecraft component testing dates back to the late 1950s, and form part of facilities ranging from complex government laboratories to small university laboratories. Schwartz et al. (2003) also note that different types of test facilities are better suited to certain applications than others, but they highlight that air bearing test facilities are well suited to experiments that require low disturbance torques.

The main goal of air bearing test facilities is to recreate spacecraft dynamics. In the case of planar air bearing test facilities, a puck-shaped air bearing is used to support vehicles on a thin pressurised film of gas. This gas film acts as a lubricant and creates a low friction interface between the vehicle and the platform atop which it floats. Planar systems allow two translational degrees of freedom and one rotational degree of freedom. (Schwartz et al., 2003).

Schwartz et al. (2003) also discuss the widespread use of spherical air bearings in so called rotational air bearing test facilities to achieve three degrees of rotational freedom. Furthermore, the most intricate air bearing test facilities combine the technologies of planar and rotational air bearing facilities to achieve systems with six degrees of freedom (Schwartz et al., 2003).

Trends in the literature suggest that air bearing test facilities see widespread use because of their robustness, comparatively low complexity, and the various types of experiments they can support. This project will focus on the development of a planar test facility, because it meets the needs of the research group.

2.1.3 Applications of planar air bearing test facilities

The nature of planar air bearing test facilities makes them well suited to running experiments related to the motion control of satellites. Rybus & Seweryn (2016) summarise some typical applications of planar air bearing test facilities, which include:

- Testing performance of free-floating robotic manipulators during orbital operations.

- Formation flying and proximity procedures of satellites.
- Testing of tethered spacecraft.
- Experiments involving the docking of spacecraft.
- Landing manoeuvres and locomotion on low-*g* bodies.
- Educational testbeds and demonstrations.

2.2 Typical features and components of planar air bearing test facilities

Rybus & Seweryn (2016) identify key parameters for air bearing test facilities. The parameters relevant for planar test facilities are:

- The design of the gas supply and air bearings.
- Whether more than three air bearings are used.
- The facility platform's surface material and construction.
- The sensors used for the test facility.

The design of the gas supply and air bearings are crucial to the operation time of the test facility. According to Rybus & Seweryn (2016), operation times range from 1 minute to 45 minutes in some cases. Typical air bearing vehicles make use of three air bearings, and to minimise volume storage pressures as high as 300 bar are used. Air bearings generally create gas films ranging from 5 µm to 15 µm in thickness depending on load and operation pressure. This gas film can reduce the friction coefficient between the bearing and the guide surface down to as low as 10^{-5} (Rybus & Seweryn, 2016).

For typical commercial air bearings with load ratings suitable for use with satellite test facilities, inlet pressures range from 2 bar to 6 bar, and air consumption per air bearing typically ranges from 0.5 NLPM to 1.5 NLPM (New Way Air Bearings, n.d.; OAV Air Bearings, n.d.; PI Nelson Air, n.d.). A schematic of a commercial planar air bearing with mounting hardware and an air inlet fitting is shown in Figure 2.1. New Way Air Bearings (2006) have published a white paper with helpful information applicable to incorporating air bearings into various designs. The porous carbon substrate allows for the gas to create a film with a uniform pressure, which improves performance of the bearing compared to the alternate use of orifices. Optimally, three air bearings should be used when supporting a simple load on a flat surface, because using four air bearings to support a stiff structure tends to have a rocking problem (New Way Air Bearings, 2006). This happens because the four air bearings are difficult to keep aligned in a plane – it is much easier to do so with three air bearings. Theoretically, using one air bearing should be possible because it can create an aligned planar interface on its own.

Additionally, the porous carbon substrate used creates a higher pitch moment stiffness compared to orifice-based counterparts (New Way Air Bearings, 2006), and so this kind of air bearing is more resistant to tipping. For this reason, the use of one air bearing instead of three will be investigated in this project.

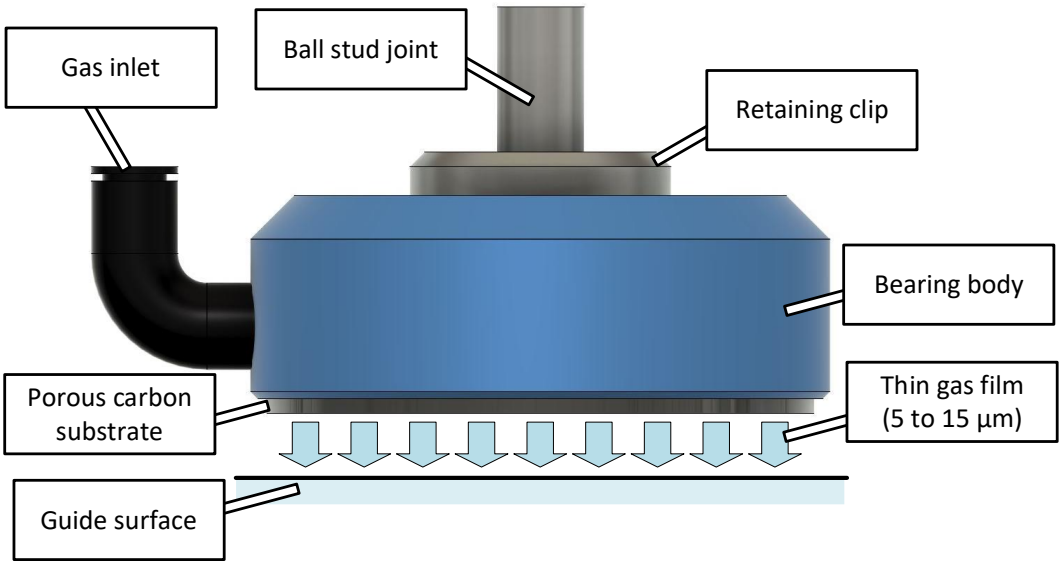


Figure 2.1 - Schematic of a planar air bearing with mounting hardware and inlet fitting.

The surface material and construction of the planar air bearing test facility's platform is important for simulating microgravity conditions, especially the surface's perpendicularity to the gravity vector (i.e. it's levelness) (Rybus & Seweryn, 2016). The roughness and flatness of the surface should be in compliance with the requirements of the air bearing as specified by manufacturers. The most common materials used for constructing the platforms are granite, glass, and epoxy (Rybus & Seweryn, 2016). Glass platforms make use of adjustable legs for levelling and are typically limited to smaller sizes than the two alternatives – but can achieve low roughness and high flatness. Granite platforms have similar properties and construction compared to glass platforms but can be built slightly larger. Epoxy surfaces can be used for the largest of platforms. These surfaces are levelled during the construction process and do not make use of adjustable legs like alternatives. However, epoxy surfaces can only achieve moderate flatness and roughness levels (Rybus & Seweryn, 2016). For this project, a glass platform with adjustable legs is ready for use.

According to Rybus & Seweryn (2016), the sensors used in modern planar air bearing test facilities can be divided into three groups:

1. Sensors designed for use on spacrecraft which are tested using the facility.

2. Sensors which are used to simulate other sensors which will be available on the spacecraft.

3. Other sensors which provide additional information regarding vehicles on the testbed, for example ground truth position measurements.

For this project, the focus will be on implementing a sensor which forms part of the third group, as the other two groups are specific to experiments and do not form part of the general test facility. In typical planar air bearing test facilities, the positions of the vehicles are measured using an overhead camera and some special markers such as light emitting diodes or fiducial markers (Rybus & Seweryn, 2016). Among commercial solutions and custom-made solutions, the precision of position measurements achievable range from a few centimetres to sub-millimetre accuracy (Rybus & Seweryn, 2016).

From the applications discussed by Rybus & Sewryn (2016), and the test facilities discussed by Schwartz et al. (2003), many of the air bearing vehicles make use of cold gas thrusters for translation and rotation. In some instances, the vehicles use momentum wheels or similar technologies for rotation along with thrusters for translation. Cold gas thrusters are convenient to use because they can potentially make use of the same gas supply which the air bearings use.

2.3 Summary

This chapter provided an overview of planar air bearing satellite test facilities. Alternative types of test facilities were summarised, but this project focuses on the planar air bearing type because of the available infrastructure and the fact that it meets the needs of the research group. The history of air bearing test facilities and the applications that the planar variant is capable of were outlined. Finally, the typical features and components of existing planar air bearing test facilities were explored. Using the knowledge of applications and solutions found in the literature, a system-level design of the test facility can now take place.

3 Test facility system-level design

3.1 Desired system characteristics

The desired characteristics of the test facility are listed below:

- The test facility should be able to be used for various satellite experiments such as docking, formation flying, debris capture and more.
- The test facility should partially mimic orbital conditions.
- An optical positioning system should be included to allow users to make use of its pose estimations as a form of ground truth measurements.
- The vehicles should be able to communicate and be controlled by wireless means.
- The vehicles should small enough such that their size is on the order of 40 cm x 40 cm x 40 cm to be compatible with and similarly sized to CubeSat components.
- The vehicles should reasonably lightweight to allow users to carry them around easily to make adjustments and refill propellant.
- The vehicles should allow for experiments to last at least 10 minutes at a 10% thrust duty cycle.
- The vehicles should have interfaces which allow users to easily expand the capabilities of the vehicle to match their requirements.
- The vehicles should be easy to assemble, maintain and modify.
- The system should be safe for users.

These are the high-level requirements of the system and drove the design of the test facility as a whole.

3.2 System architecture

Figure 3.1 shows the architecture of the test facility, highlighting the main subsystems, their components, and their interactions with each other. This thesis handles the design of the feedback and communication subsystem, carrier cart subsystem and propellant refill subsystem. The glass platform was already available for use at the start of the project.

The feedback and communication subsystem is responsible for determining, displaying, and logging pose measurements (and other data) of the experiments done on the test facility. This subsystem also allows the user to input parameters

related to the pose estimation and the experiment itself. Additionally, it is capable of wirelessly communicating the pose measurements and input parameters to the payload computer on the carrier cart in near real-time. The communication network is capable of bi-directional communication between itself and the payload computer of the carrier cart, allowing for further information from the payload to be logged and displayed for the user.

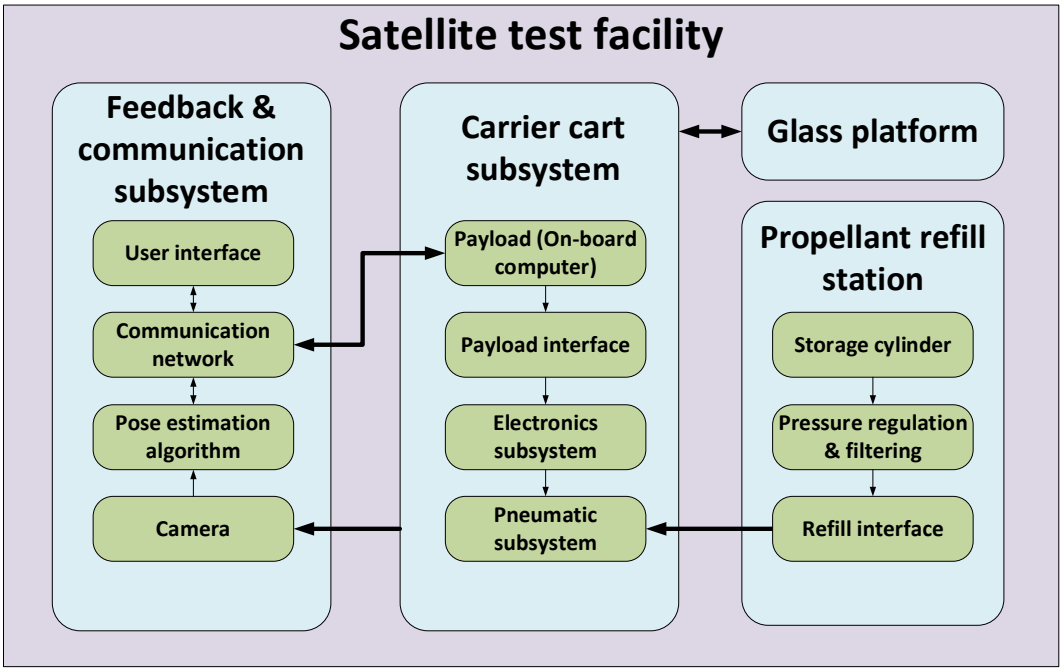


Figure 3.1 - Architecture of the satellite test facility.

The carrier cart subsystem consists of the air bearing vehicles of the test facility, named carrier carts, which emulate satellite motion by providing low-friction planar movement on the glass platform. The carrier cart makes use of a pneumatic subsystem to provide propellant for the thrusters which move it, and the air bearing which allows it to hover on the glass platform. The test facility is designed to handle multiple carrier carts and provide pose measurements and feedback for each. The carrier cart is designed to carry and interface with a payload which consists of an on-board computer and any other components which future users need for their experiment. A demonstration payload will be designed to demonstrate the test facility's capabilities.

The propellant refill station's purpose is simply to refill the carrier cart's gas supply between experiment runs. The refill process is designed to be safe and fast for the convenience of users.

3.3 Functional description

Figure 3.2 describes the general functioning of the test facility during an experiment. The user needs to design their payload to interface mechanically and electrically with the carrier cart. If desired, the payload can also be designed to interface with the communications network to send and receive information from the feedback and communication system. The payload can control the motion of the cart and receive feedback according to the needs of the experiment.

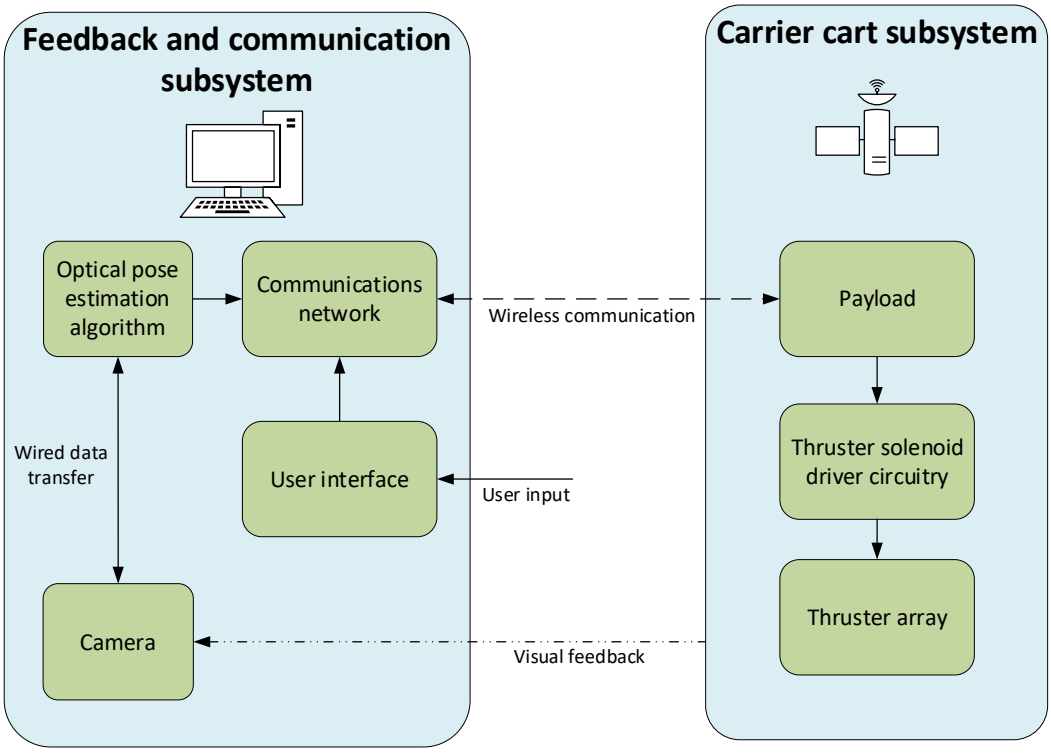


Figure 3.2 - Functional architecture of the test facility during an experiment.

The feedback and communication system is run on a desktop computer and acts as a control panel for the test facility, allowing the user to interface with their experiment and the facility. It allows for visualising of current data, logging of data and changing the parameters of the feedback and communication system. The user can also use it to send updated parameter values and other data to their payload as well as receive, display, and log relevant data from their payload.

Figure 3.3 shows a conceptual sketch of the physical layout of the test facility during operation. The key subsystems and components are also indicated. Not shown is the propellant refill station, which is not used during operations but rather between experiments.

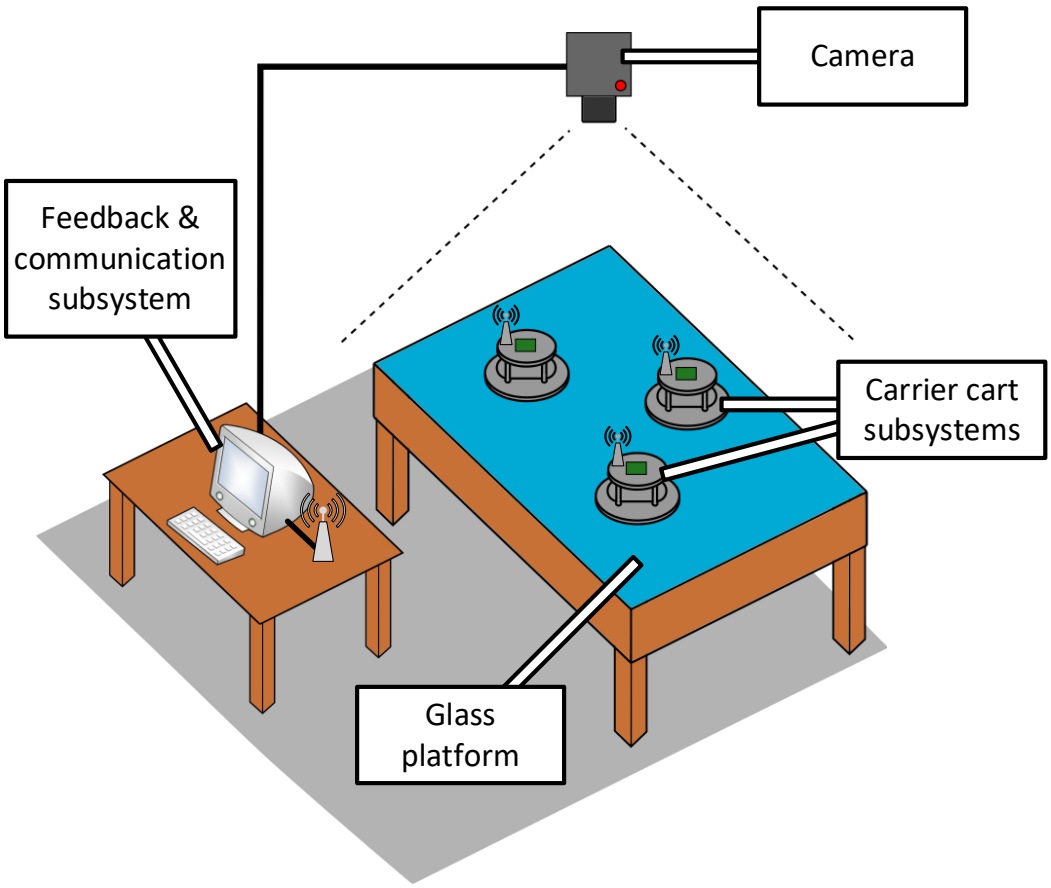


Figure 3.3 - Conceptual sketch of the test facility during operation.

3.4 Summary

This chapter discussed the design of the satellite test facility on a system level. The desired characteristics were listed, after which the system architecture was described. This was followed by a functional description of the test facility during a typical experiment.

4 Hardware design

This chapter discusses the design of the hardware subsystems of the test facility, namely the carrier cart subsystem, the propellant refill station, and the demonstration payload's circuitry. The focus is on the carrier cart, which made up the bulk of the hardware design.

4.1 Carrier cart design requirements

The main requirements of the carrier cart are as follows:

- The cart should have an overall size smaller than 400x400x400 mm.
- The cart should have a mass of less than 10 kg.
- The cart's run time on the table should be more than 10 minutes at a single thruster duty cycle of 10% at a thrust level of 0.4 N.

Along with these technical requirements, it is desirable that the cart be easy to assemble and maintain and safe to users even in the event of human error, as far as possible. Another desired trait is for the cart to be versatile in the options that it provides to users to reconfigure, tune, and make additions for their requirements.

4.2 Design process for the carrier cart

The design methodology for the carrier cart followed the design process as outlined by Dieter & Schmidt (2013). This began with an analysis of the customer requirements, that is, the requirements of the cart to achieve success according to the problem statement. These customer requirements were then translated into engineering requirements, giving a list of desired specifications to aim for. This was followed by constructing a house of qualities (quality function deployment) to determine the relative importance of each requirement, which assisted in decisions about design trade-offs. The house of qualities is shown in Appendix A.1.

After the specifications of the design were set, functional and morphological decompositions of the carrier cart were performed. These decompositions aided in detailing the functioning required of the cart and highlighting the types of components needed. Following this, the theory of inventive problem solving (TRIZ) toolset was used to attempt to improve on the previous air bearing vehicle.

Once the specifications and functions of the carrier cart were determined, several concepts were generated. Three concept sketches are shown in Appendix A.2. The concepts were evaluated using a weighted decision matrix. From this, a final

concept was selected and formed the basis for the embodiment and detail design of the carrier cart.

The embodiment design process involved many iterations of selecting different sets of suitable components from different manufacturers. Each set of components was selected such that the design specifications were met. A comparison was then done between all the suitable sets of components, where evaluation criteria were cost, size and complexity of assembly and maintenance. Based on this evaluation, a final choice was then made on which set of components would be used for the detail design.

The detail design consisted of designing the specific interfaces between components and designing the final assembly of the carrier cart. This included compiling drawing packs for machined components and compiling bills of materials for the main assembly and subassemblies.

4.3 Carrier cart architecture

Figure 4.1 shows the architecture of the carrier cart, with the interfaces between subsystems outlined. The main constituents of the carrier cart are the pneumatic subsystem, the electronics subsystem, the structural subsystem, and the payload interface.

The pneumatic subsystem handles the storage and distribution of pressurised gas necessary for the cart to function. The pressurised gas is supplied to both the air bearing and the propulsion array. The air bearing provides the low-friction load-bearing interface between the carrier cart and the glass platform. A group of cold gas thrusters make up the propulsion array which allows the carrier cart to move around on the glass platform. The glass platform has a usable area of 3.2x1.8 m.

The electronic subsystem is responsible for providing electrical power to the payload and driving the thruster valves. The power delivery circuit regulates voltage levels and provides electrical power to both the payload and the thruster valve driving circuitry. The driving circuitry is responsible for conditioning control signals from the payload and sending these conditioned signals to the electronically controlled thruster valves.

The structural subsystem simply acts as a chassis, providing mechanical support and allowing for mounting of all the components of the carrier cart. It consists of support plates, support columns and various brackets and fixtures to achieve this function.

The payload interface consists of both a mechanical and electrical interface, which allow the payload to interact with the carrier cart. The electrical interface transfers

electrical power to the payload and allows for the payload to control the cart's thrusters by sending the payload's control signal to the thruster driving circuitry. The mechanical interface simply provides mechanical support and mounting points for the payload.

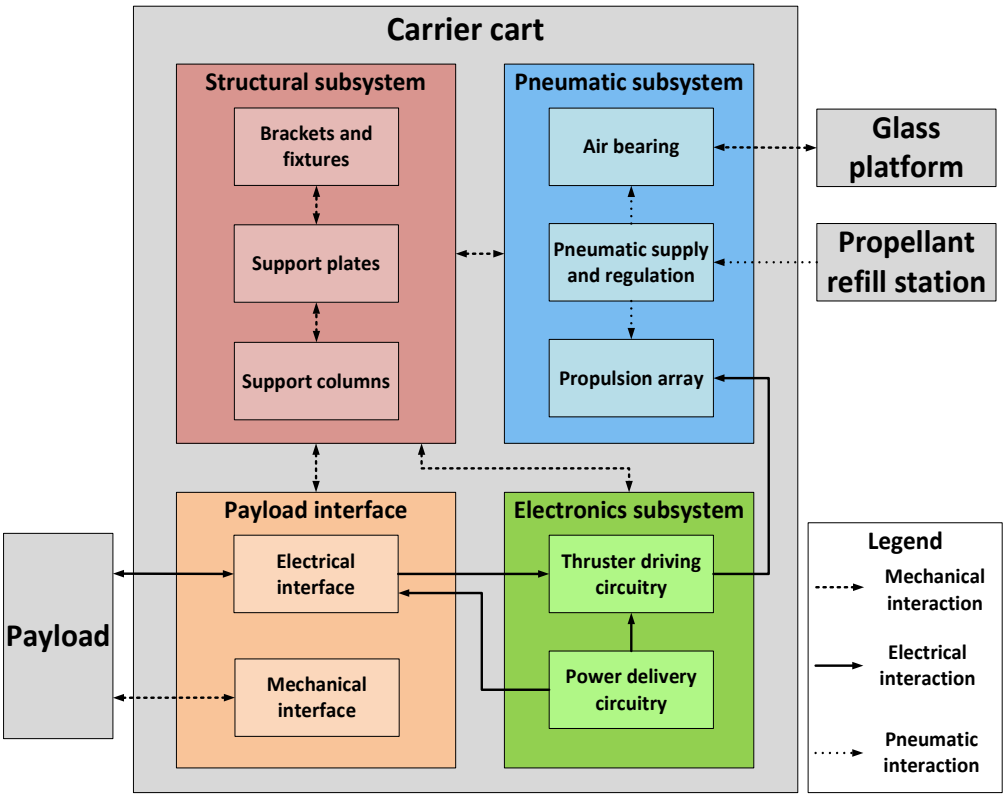


Figure 4.1 - The architecture of the carrier cart, showing interactions between itself and other subsystems of the test facility, as well as interactions between its own subcomponents.

4.4 Carrier cart detail design

4.4.1 Overview of the carrier cart

Figure 4.2 and Figure 4.3 show photographs of the front and left view of the assembled carrier cart and annotates some of the key subsystems and components on the cart. The cart was designed to allow easy assembly, modification, and maintenance by future users. The assembled cart has a mass of 7.5 kg and an overall size of 330x338x204 mm, and so two of the main requirements of the carrier cart are met.

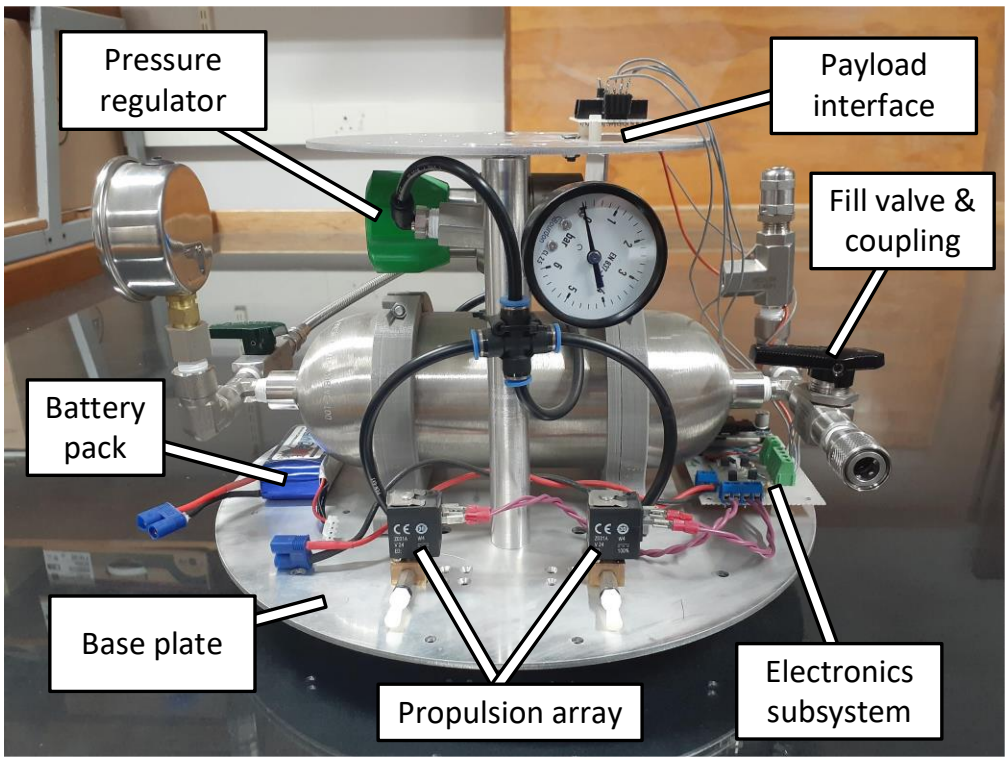


Figure 4.2 - Photograph of the front view of the assembled carrier cart, highlighting some key constituents of the carrier cart.

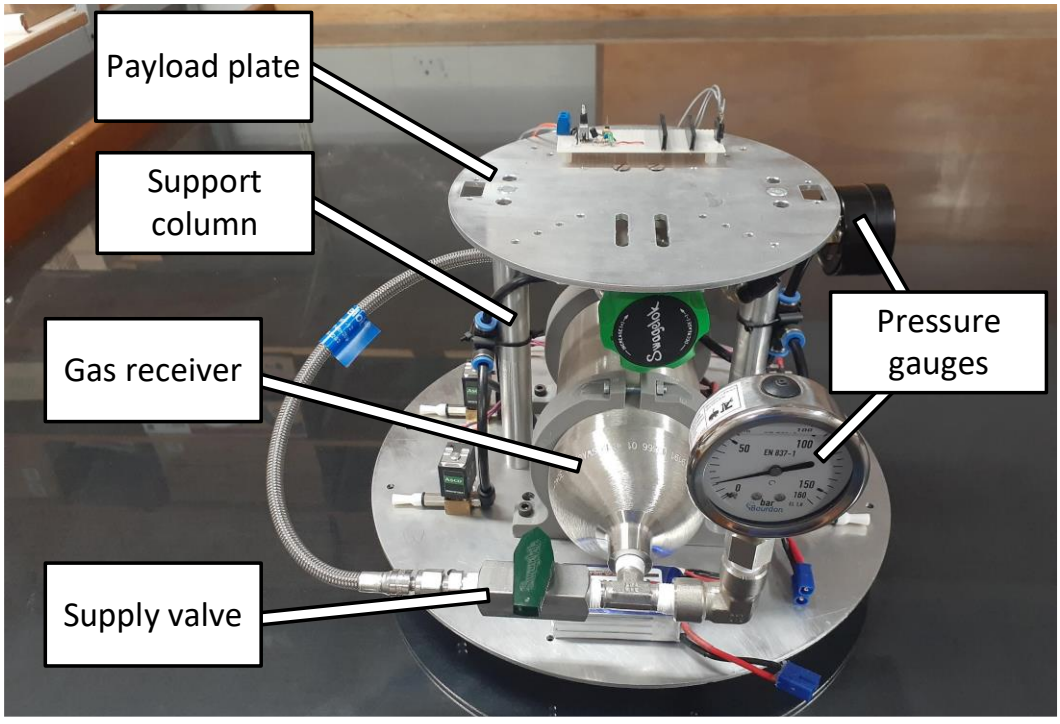


Figure 4.3 - Photograph of the left view of the assembled carrier cart, highlighting some more key constituents of the carrier cart.

In theory, the designed cart achieves a runtime of 14 minutes and 35 seconds for a thrust level of 0.4 N at a 10 % duty cycle for a single thruster. This theoretical runtime is calculated assuming that the thrusters of the cart can achieve a commercially obtainable specific impulse of 40 s. So, in theory, the cart achieves the third main requirement of the cart. Chapter 5 investigates the performance of the thrusters of the carrier cart in more detail. The following subsections discuss the details of the design of each of the carrier cart's subsystems and their components.

4.4.2 Pneumatic subsystem design

The pneumatic subsystem is responsible for receiving, storing, and supplying gas to the thrusters and air bearings on the carrier cart. Nitrogen is chosen as a working gas because of its availability, compatibility with air-certified components and its slightly better performance as a propellant. It also poses little safety hazards to users during regular use of the test facility. Nitrogen is used for both the air bearing and thruster array (as opposed to two different gasses) to reduce the cost, complexity, and size of the system.

Figure 4.4 shows the pneumatic circuit diagram describing the configuration of the components which make up the pneumatic subsystem. Most of the components contribute to the supply and regulation of nitrogen gas, which then serves the working components – the air bearing and the thruster array. All of the pneumatic components are chosen to be compatible with the nitrogen medium and to be corrosion resistant. The interfaces between threaded components are also chosen such that minimal galvanic corrosion occurs between dissimilar metals in contact.

The receiver and relief valve ensure that pressurised nitrogen is stored safely for use by the system. A receiver size of 1 L and a rated pressure of 124 bar is chosen to store a sufficient mass of nitrogen to run the experiments for the required amount of time. When full, the receiver holds approximately 145 g of nitrogen gas. As a safety precaution both to the components and the user, the relief valve ensures that the storage pressure does not exceed the rated pressure in the case of human error during filling.

The fill valve and quick coupling make up the interface to the propellant refill station, allowing for the receiver to be refilled with nitrogen. The fill valve is a manual ball valve actuated by the user during the refilling process. A ball valve is chosen for its ability to resist the back pressure of the gas stored in the tank. The quick coupling is simply a ball detent style coupling to allow for easy connecting and disconnecting from the refill station. Another safety feature of the design is that the quick coupling does not allow uncoupling under pressure, ensuring that the fill valve must be closed before uncoupling.

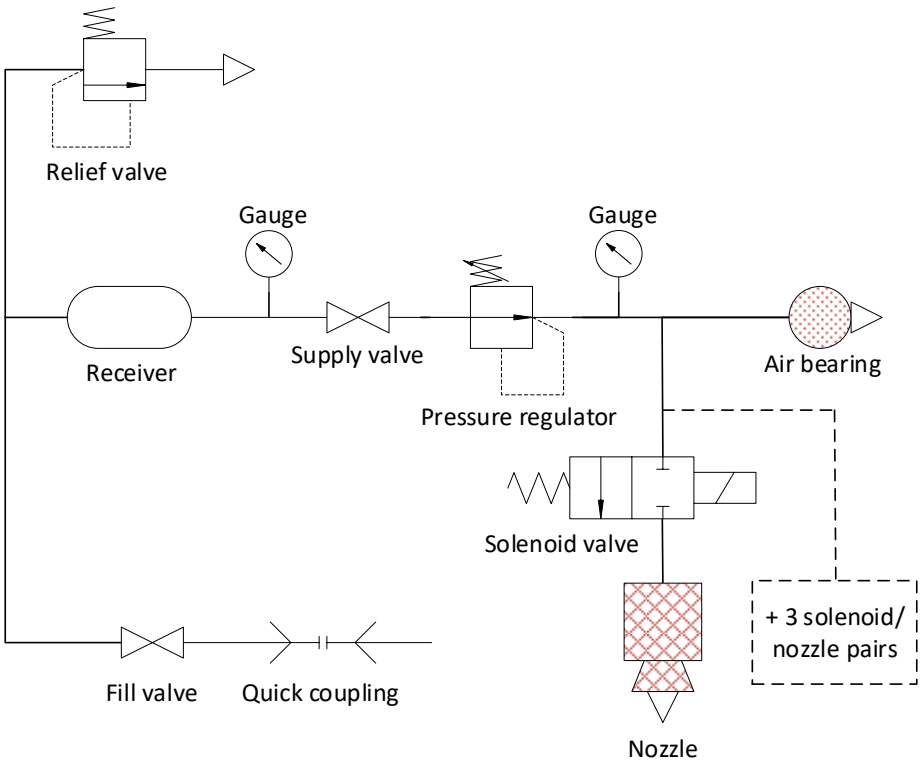


Figure 4.4 - Pneumatic circuit diagram describing the layout of the pneumatic subsystem on the carrier cart.

The nitrogen supply to the working components is enabled via the supply valve, after which the nitrogen's pressure is regulated to the desired working pressure of 4.5 bar. The choice of 4.5 bar as a working pressure is related to the air bearing's rated working pressure range, as well as the selection of solenoid valves as discussed in Subsection 4.4.3. The supply valve is a manual plug valve which should be actuated by the user to enable the gas supply to reach the working components. A plug valve is chosen since it does not need to resist back pressure and is a lower cost alternative to a ball valve. The pressure regulator is an adjustable spring-loaded pressure reducing regulator with an output range of 0 to 6.7 bar. The selection of output range and components ensures that no downstream components are damaged due to human error during setting of the output. A filtered regulator is chosen to minimise the risk of small particulates reaching the air bearing or thrusters, which could inhibit their functioning. The chosen regulator's droop (outlet pressure variation due to flow rate) stays within acceptable levels for the expected flow rates as long as the inlet pressure remains above 10 bar. For this reason, the carrier cart is to be considered empty when its tank pressure is lower than 10 bar.

The pressure gauges on the up- and downstream sides of the pressure regulator indicate the storage pressure and working pressure respectively. This allows the

user to see the amount of remaining propellant and to receive feedback when adjusting the pressure regulator to working pressure.

The carrier cart is designed to use either one or three air bearings. It was found that one air bearing is sufficient to support the carrier cart during normal motion where the cart does not interact with other objects. The allowance for three air bearings was made for the case where experiments may require the carrier cart to interact with other carts or objects, where large lateral loads are expected. The larger lateral loads are more likely to cause the single air bearing to tip and introduce unwanted friction. The three air bearings will create a more stable planar interface necessary in these cases but will use more gas and will shorten the experiment duration. The air bearing has an ideal load rating of 222 N, where the gas film has a thickness of 5 µm. For the case of the carrier cart, with a weight of 74 N, a film thickness of about 12.5 µm is theoretically achieved. A higher film thickness leads to higher gas consumption by the air bearing, so using three air bearings would significantly increase gas consumption. However, most of the gas consumption during a typical experiment is attributed to the thruster array. For perspective, the cart can theoretically hover for at least 111 minutes and at least 37 minutes for one and three air bearings respectively, assuming the thrusters are not used.

The thruster array consists of four thrusters, where a single thruster is made up of a solenoid valve and nozzle pair. The details of the design of the thruster array is discussed separately in Subsection 4.4.3.

The conduits used for transporting the fluid between components are sized to achieve a suitable maximum flow velocity and maximum pressure drop. A maximum flow velocity of less than 25 m/s is aimed for to minimise flow vibrations. A maximum pressure drop of less than 0.1 bar between major components is aimed for to ensure that the working pressure remains consistent enough during thruster use. All conduits theoretically achieve these values, and the maximum pressure drop measured in practice was found to be less than 0.1 bar.

As a safety feature, all components used in the pneumatic subsystem are rated for well above the maximum pressures that they will see in the worst-case scenario, even if human error is introduced during either the refilling or regulator setting process. A full list of the specific parts used for the pneumatic subsystem can be found in the bill of materials in Appendix A.3.

4.4.3 Propulsion array design

The propulsion array is a group of thrusters which can be used to move the carrier cart around on the glass platform. The first important design consideration was the configuration of the thrusters, as this determines (among other things) the

number of thrusters required and thus the amount of nitrogen propellant which will be used during operation. Figure 4.5 shows some of the thruster configurations considered during the initial design phase. The thrusters are labelled with N, the centre of mass is labelled as CoM, and reaction wheels are labelled with RW. Parameters L and φ are the offset from the centre of mass and the pivot angle respectively. Also shown is the x-y axis system of the body of the carrier cart.

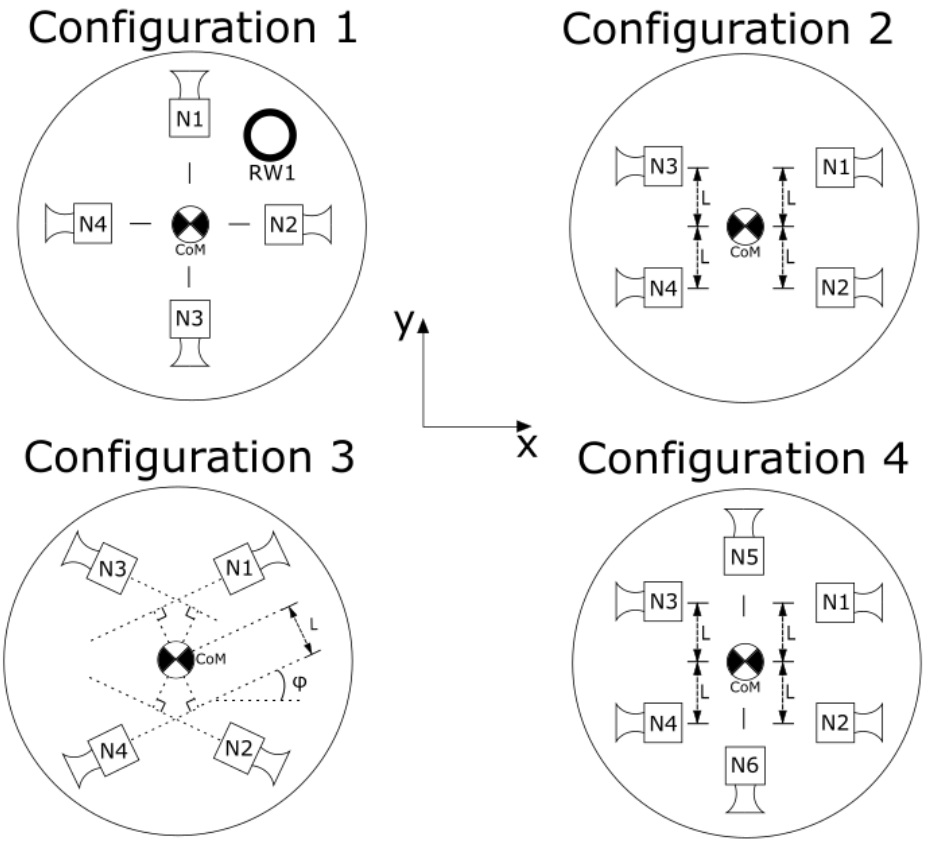


Figure 4.5 - Different thruster configurations considered during the design of the propulsion array.

Each of these configurations can achieve translation without rotating and vice versa by the activation of certain thrusters in pairs. This is desirable to ease the control of the cart.

Configuration 1 features four thrusters with their lines of action going through the centre of mass of the cart. This configuration allows for translation in both the x and y direction of the carrier cart body, but the rotation of the cart cannot be controlled by the thrusters. A reaction wheel or something similar would be necessary alongside the thrusters for this purpose. This configuration would use

low amounts of propellant, but requires an extra level of complexity since another device needs to be used for rotation control.

Configuration 2 makes use of four thrusters with their lines of action all parallel to the x axis, but offset from the centre of mass. This allows for rotational and translational control, but the translation can only occur in the x direction of the carrier cart body. This configuration will use more propellant than configuration 1, but does not require another device to perform rotations. However, this is at the cost of losing the ability to translate in the y direction of the carrier cart body.

Configuration 3 is similar to configuration 2, aside from the fact that the thrusters are pivoted to allow for motion in the y direction of the cart body. This added capability comes at the cost of generally more thrust usage, and more inefficient usage of propellant. It is more inefficient because when thrusting in the x or y direction, each thruster wastes propellant by counteracting the paired thruster's motion in the perpendicular direction. For example, if the cart is commanded to translate in the positive y direction, thrusters 2 and 4 should be activated – and in this case the thrust created in the x direction from both thrusters is wasted, increasing propellant usage per unit of thrust in the y direction. This decreases the effective specific impulse of the thrusters.

Configuration 4 uses six thrusters to allow for rotation and translation in both the x and y direction, while negating the inefficiencies induced by pivoted thrusters. The use of six thrusters does however add to the complexity of the pneumatic supply system and encroaches available space on the cart, not to mention the increased cost.

The default configuration used in the final design is configuration 2, as it achieves a good balance of propellant use, complexity of supply, and efficiency of propellant use. The downside of this configuration is that the cart can only accelerate in the x direction of the cart body. However, since some experiments may require translation in both the x and y direction, the mounting interface of the thrusters is designed to allow for brackets to be used to add pivot angles to the thrusters. This allows for configuration 3 to be achieved with and adjustable pivot angle if desired by the user.

As mentioned previously, the type of thruster used on the cart is a cold gas thruster. Cold gas thrusters are one of the simplest types of rocket engines and use a pressurised gas as a propellant. No combustion occurs in a cold gas thruster's chamber. This type of thruster is convenient to use on the cart since it simply makes use of a pressurised inert gas as its propellant, similar to that required by the air bearing. The thrusters on the cart consist of a solenoid valve and nozzle pair. Figure 4.6 shows an exploded-view drawing of the components that make up a thruster subassembly on the cart.

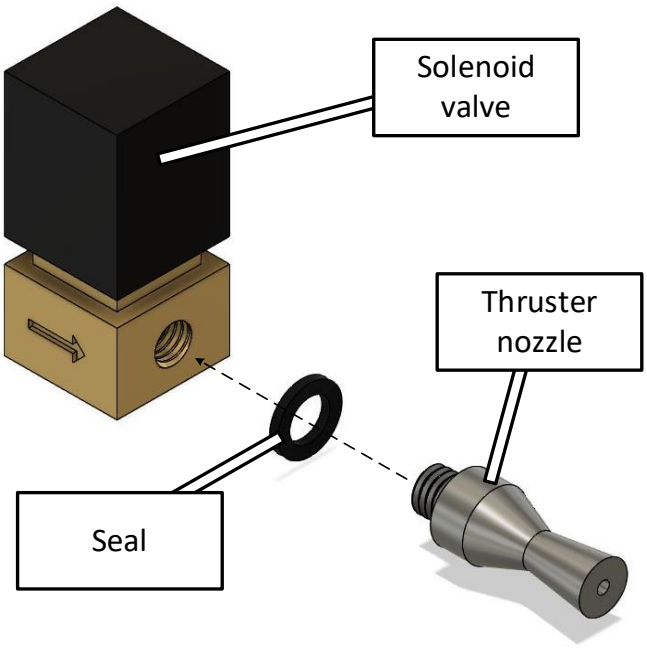


Figure 4.6 - Exploded-view drawing of the components that make up a single thruster on the carrier cart.

The geometric profile of the thruster nozzle influences the force output and the specific impulse of the thruster. The specific impulse of a thruster is a measure of its efficiency in propellant usage and is the ratio of the force output divided by the weight of the propellant flowing through the nozzle each second. For the carrier cart, an acceleration of around 0.1 m/s^2 is desired. Given the thruster configuration, and assuming a payload mass of 500 g, the nozzle is required to output a force of 0.4 N. To achieve the third main requirement of the cart, a runtime of 10 minutes at a thrust duty cycle of 10% is needed. This corresponds to the thruster nozzle requiring a specific impulse of 24 s or higher.

The procurement of the nozzle required proved to be a problem. The geometric profiles and feature sizes required by the nozzle made it very difficult to have it machined locally. Commercial options were available from foreign suppliers, but they had very high costs. For this reason, it was decided to investigate the feasibility of 3D printed nozzles and to characterise their performance. Chapter 5 discusses the design and manufacturing of the 3D printed nozzles in detail.

The main specifications driving the selection of the solenoid valves are the opening/closing time of the valve and the orifice size. The opening/closing time directly affects the minimum impulse bit of the thruster, which is the minimum impulse that the thruster can apply to the carrier cart. To be comparable to commercial solutions, a minimum impulse bit of 5 mN·s is desired. This requires a

valve opening/closing time of 12.5 ms or less. The solenoid valve used in the design has a specified opening/closing time of 10 ms or less and was measured to have an opening/closing time of 9 ms. Details of the measurement are discussed in Chapter 5. The orifice size of the valve is related to the flow coefficient, and thus to the pressure drop across the valve when it is open. Given the expected flow rates and a desired pressure drop of 0.5 bar or less, a flow coefficient of 0.087 or less is required. The solenoid valve used in the design has a specified flow coefficient of 0.07. An alternative consideration was to use a proportional valve instead of a solenoid valve, which would provide analogue control of the flow rate. However, suitable options were found to be comparatively much more expensive and had limited availability locally.

More details on the nozzle design and the performance of the thruster can be found in Chapter 5.

4.4.4 Electronics subsystem design

As discussed previously, the electronics subsystem consists of a power delivery circuit, as well as a thruster driver circuit which drives the solenoids of the thruster array. The power delivery circuit is shown in Figure 4.7.

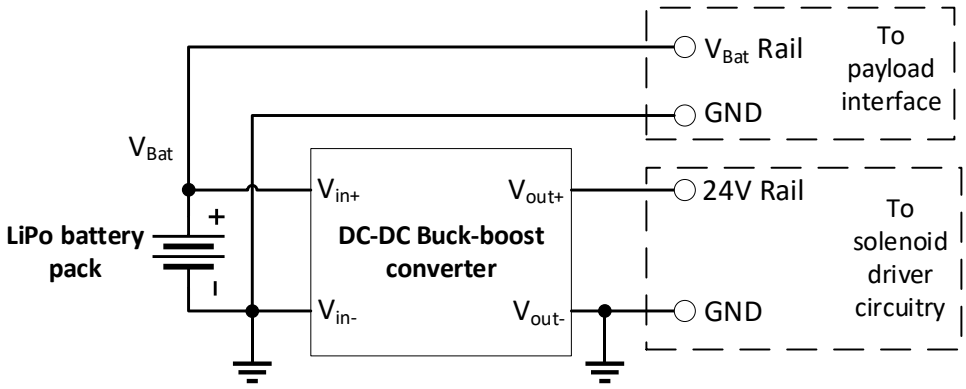


Figure 4.7 - Diagram describing the layout of the power delivery circuit of the carrier cart's electronics subsystem.

The power delivery circuit consists of a rechargeable lithium-polymer (LiPo) battery pack and a variable DC-DC buck-boost converter. The battery is directly connected to the payload interface to provide power to the payload, which is required to step down and regulate the voltage if necessary. A battery voltage of around 12 V is desired to allow for use with most 5 V and 3 V regulators which will typically be used for common microcontrollers. Consequently, a 3-cell LiPo battery pack is used to achieve a nominal voltage of 11.1 V. In the case where the payload needs higher than 12 V, a LiPo battery with more cells can be used and the converter can be adjusted accordingly. This is the reason a buck-boost converter

is used instead of a boost converter. The converter's role is simply to step up the battery voltage to the 24 V required by the solenoid valves of the thruster array.

The thruster driver circuit consists of an array of four identical solenoid-driving subcircuits. Each subcircuit receives a binary control signal from the payload, indicating whether the solenoid valve it controls should be opened or closed. The role of the driver circuit is to protect the payload from damage and to condition the control signal. Figure 4.8 shows a single solenoid-driving subcircuit.

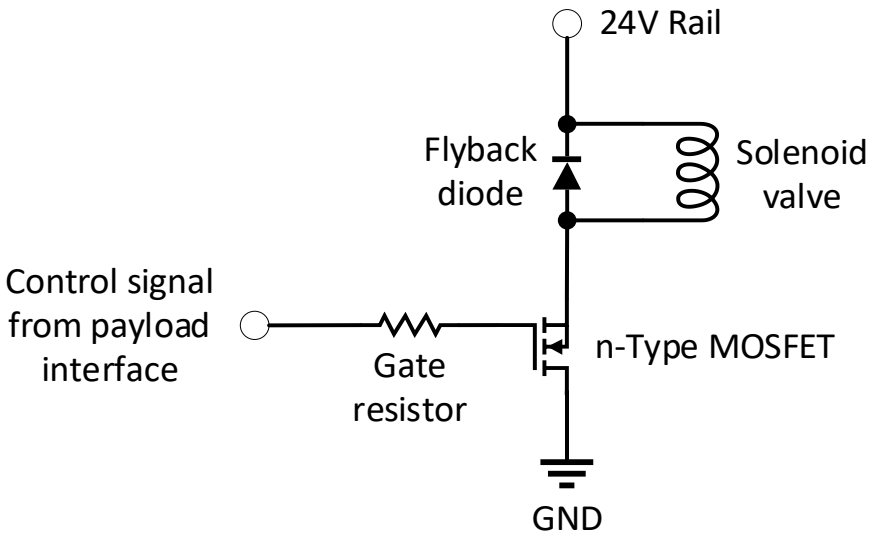


Figure 4.8 - Circuit diagram describing the layout of a single solenoid-driving circuit.

The driver circuit uses an n-type MOSFET to switch the state of the load using the control signal from the payload. Since the solenoid valve is a highly inductive load, a flyback diode is used to prevent damage from inductive kickback when the solenoid is switched off. The solenoid is powered through the 24 V rail, which is the output form the DC-DC converter. The ground of the driver circuit is connected to the negative output of the converter, as well as the ground of the payload.

An integrated circuit solenoid driver was considered, but the cost and availability of such drivers negated the potential small gains in performance. The solenoid-driving circuit shown above performs acceptably and achieves the desired opening and closing time specification of less than 10 ms. The measurement of the opening and closing time is discussed in Chapter 5.

4.4.5 Structural subsystem design

The role of the structural subsystem is to provide mechanical support and mounting points for the components of the carrier cart, as well as providing means

for some adjustability for the user to mechanically tune some parameters. Figure 4.9 shows an overview of the parts that make up the structural subsystem, as well as their positioning in the carrier cart assembly.

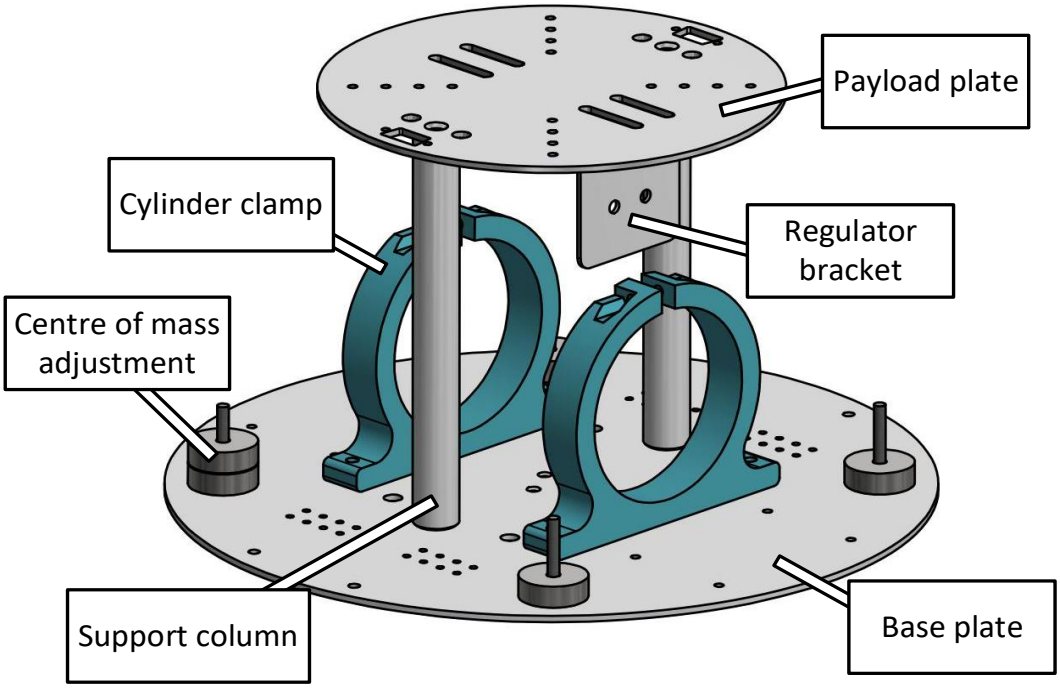


Figure 4.9 - A sketch of the parts which make up the structural subsystem of the carrier cart.

The base plate serves as the mounting plate and support for most of the carrier cart components. It also acts as the load bearing interface for the air bearing, as the bearing is mounted underneath the base plate. As mentioned before, the cart can accommodate for a single air bearing or three air bearings. For a single air bearing, the bearing is mounted directly to the base plate. For the case of three air bearings, ball stud joints are used to mount the bearings to provide them with some rotational freedom to self-align. In this case retaining clips are used to secure the ball stud joints to the air bearings.

The base plate includes routing holes for the fluid conduits that connect to the air bearing in the case of one and three air bearings. Furthermore, the base plate provides a mounting interface for the electronics subsystem, the cylinder clamps, the centre of mass adjustment points, and the thruster array. The thruster array has various mounting points, so that the moment arm between thrusters can be adjusted. Using mounting brackets, the height and pivot angles of the thrusters can also be adjusted. This adds further versatility to the carrier cart and allows users to experiment with different configurations.

The support columns simply act as standoffs between plates on the carrier cart and provide support to the layer of components mounted on the plate above them. The columns are designed to be rigid enough to mitigate vibration during thruster usage.

The payload plate provides support for the payload and acts as a mounting interface for it, offering various mounting holes for users. It also features routing holes for cables and fluid conduits if necessary. The mounting holes on the payload plate also make it possible to stack further support plates above it using the support columns as standoffs. This may be necessary for some payloads and is necessary to mount the marker plate as will be discussed in Chapter 6.

The regulator bracket and cylinder clamps simply act as mounting fixtures for parts of the pneumatic subsystem, namely the gas receiver and the pressure regulator. The rest of the pneumatic subsystem is attached to these two parts through pipe fittings and fluid conduits.

The carrier cart features four centre of mass adjustment points, which each consist of a stud attached to the base plate. Each stud is positioned equidistant from the centre of the base plate, where the centre of mass is ideally located. These studs allow the user to adjust the centre of mass by placing ring-shaped weights onto them, compensating for any eccentric masses that their payloads or other additions introduce.

4.4.6 Payload interface

The payload interface allows the interaction between the payload and the carrier cart. The interface includes a mechanical part and an electrical part.

The mechanical interface consists of the mounting points available for the payload, as well as the optional routing points for providing the payload with nitrogen gas. Both of these form part of the payload plate.

The electrical interface consists of a bundle of electrical wires which are necessary for power delivery to the payload, as well as for control of the thruster array. Figure 4.10 describes the connections made between the carrier cart electronics and the payload through the payload interface. The payload plate facilitates the mounting of a DB9 connector which can be used for making the connections quickly.

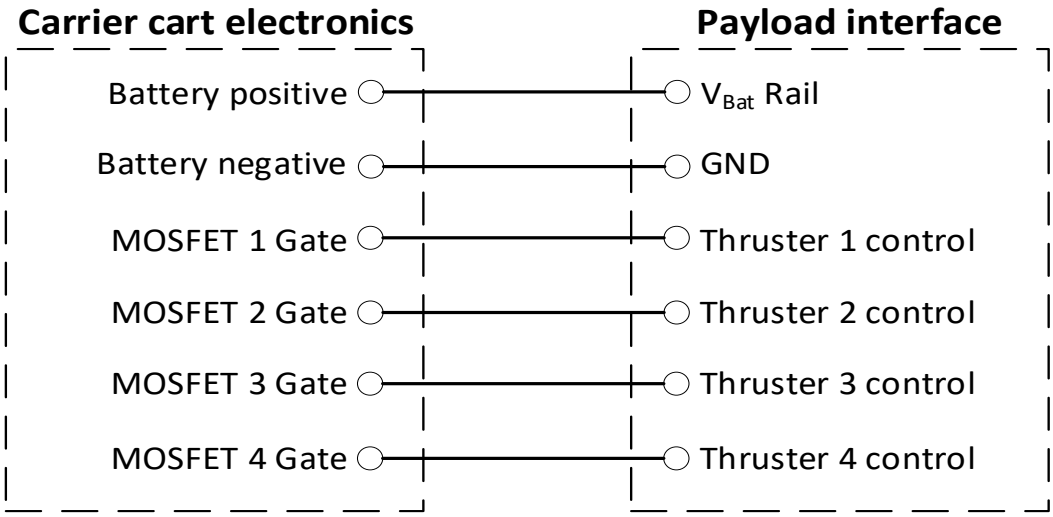


Figure 4.10 - Diagram showing the connections between the carrier cart electronics subsystem and the payload interface.

4.4.7 System modelling during the design phase

Due to the complex nature of gas flow in a system, a Simulink model was created to simulate the behaviour of the carrier cart's pneumatic subsystem during operation. The simulation considers the effects of the compressibility of nitrogen, the temperature drops due to the Joule-Thompson effect, heat transfer into the system and the effect of the temperature variations on the gas consumption of the air bearing and thrusters. Figure 4.11 shows a block diagram describing the calculations performed during the simulation. The results of the simulation were used to determine the performance of the pneumatic subsystem, including the runtime of the carrier cart. This was used to size the receiver and choose design pressures. Furthermore, this simulation was used to determine gas temperatures in throughout the pneumatic subsystem during operation. Knowledge of the approximate temperatures expected during operation was used to select suitable components capable of withstanding said temperatures.

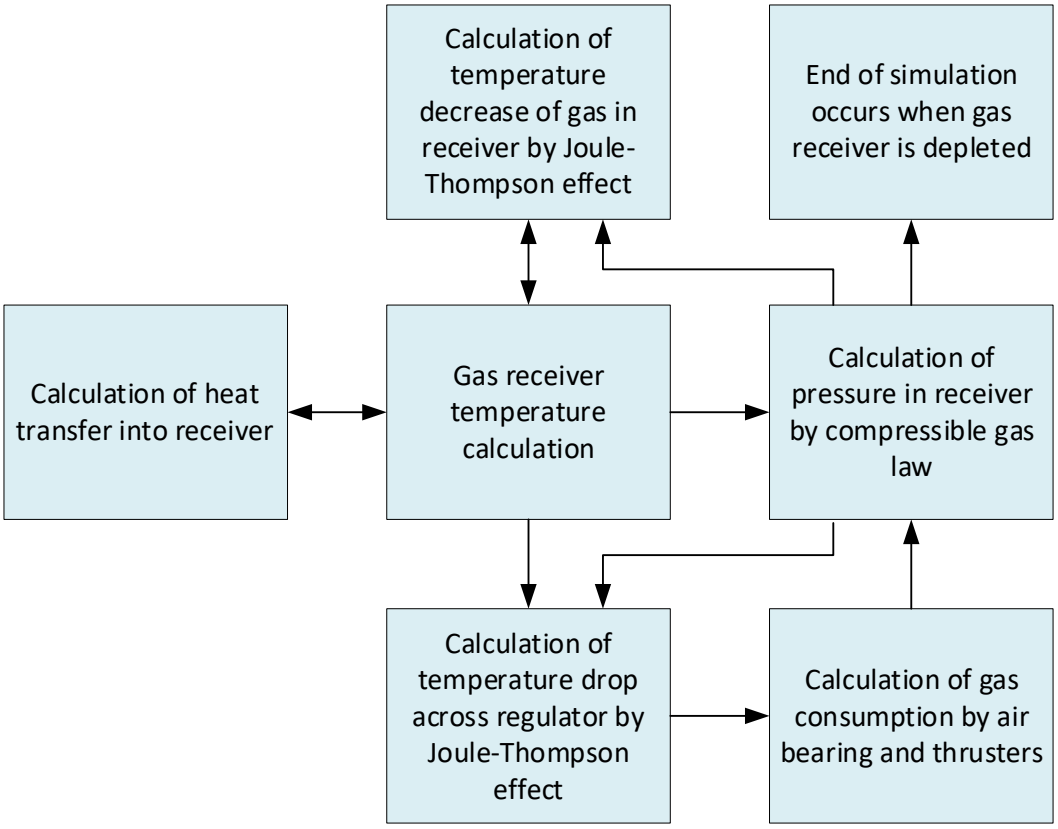


Figure 4.11 - Block diagram describing the calculations performed in the simulation of the pneumatic subsystem's operation.

Another design aspect which required many iterative calculations was the sizing of the conduits of the pneumatic subsystem. The sizing was based on the pressure drops over the conduit segments and the pneumatic components. The calculations were done using the methods outlined by Çengel & Cimbala (2014). Friction factors were determined iteratively using the Colebrook-White equation, while flow coefficients of components were determined from respective manufacturer's datasheets. These calculations were repeated for each design iteration in the embodiment design phase to evaluate and compare the performance of each iteration.

4.5 Propellant refill system design

The propellant refill system is a supporting subsystem of the test facility, which has the role of allowing users to refill the gas supply of the carrier cart quickly and easily. The pneumatic circuit diagram of the propellant refill station is shown in Figure 4.12.

A 50 L industrial gas cylinder is used for the large gas reservoir and cylinder valve pair. This cylinder is rated to 200 bar and when full it stores approximately 11 kg of nitrogen, which is used to refill the carrier cart gas supply before experiments. Once the industrial gas cylinder is empty, it is refilled externally by a company. One full cylinder can provide enough nitrogen to refill the carrier cart approximately 75 times.

An adjustable dual-stage spring-loaded pressure reducing regulator is used to regulate the pressure to 124 bar and lower when refilling the cart. A dual-stage regulator is used because of the high flow rates seen during the refilling process. The regulator includes a filter on the input to minimise the risk of particulates being introduced into the carrier cart's pneumatic subsystem. The regulator is rated for input pressures up to 413 bar and outputs pressures in the range of 0 to 137 bar. Since the regulator can output up to 137 bar, which is above the rated pressure for the carrier cart gas receiver, human error can cause damage during the refill process. This risk is however mitigated by the inclusion of a relief valve on the carrier cart.

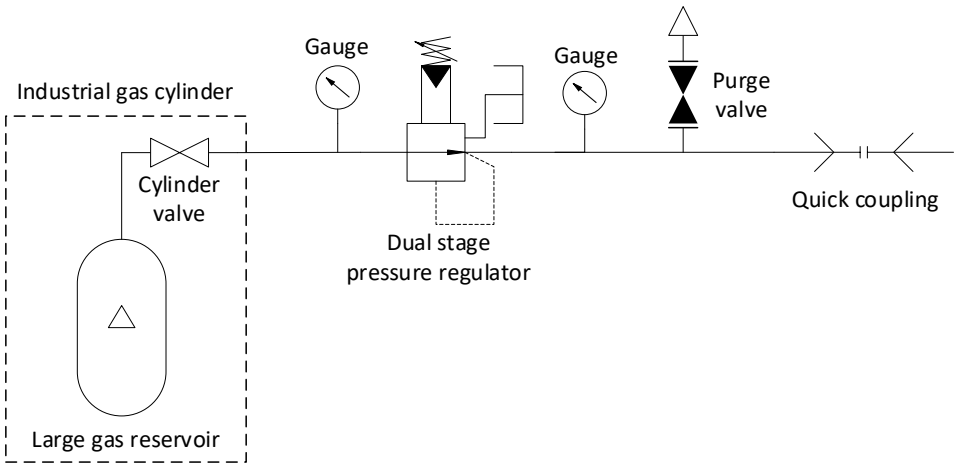


Figure 4.12 - Pneumatic circuit diagram of the propellant refill system.

Pressure gauges are added on the up- and downstream nodes of the pressure regulator to allow the user to read off the cylinder pressure and the fill pressure. This facilitates checking of the remaining gas left in the cylinder, as well as providing feedback for setting the fill pressure correctly.

A purge valve is a necessary addition to allow the user to bleed off excess gas from the fluid conduits after refilling the cart and closing all other valves. This is required to depressurise the conduits and to disconnect the carrier cart coupling.

The quick coupling allows for the convenient connection of the carrier cart's pneumatic subsystem to the refill system. The stem of the quick coupling is

attached to a long flexible hose attached to the output node of the regulator. This long hose allows for an easy connection to the carrier cart which should be positioned nearby when refilling. The body of the quick coupling is present on the carrier cart.

Two annotated photographs of the propellant refill system assembly are shown in Figure 4.13. The pressure regulator is connected to the industrial cylinder using a standard bullnose connector for nitrogen gas. The carrier cart storage stand provides support for the carrier cart without damaging the air bearing mounted underneath, and is also used for storing the carrier carts on a surface when not in use.

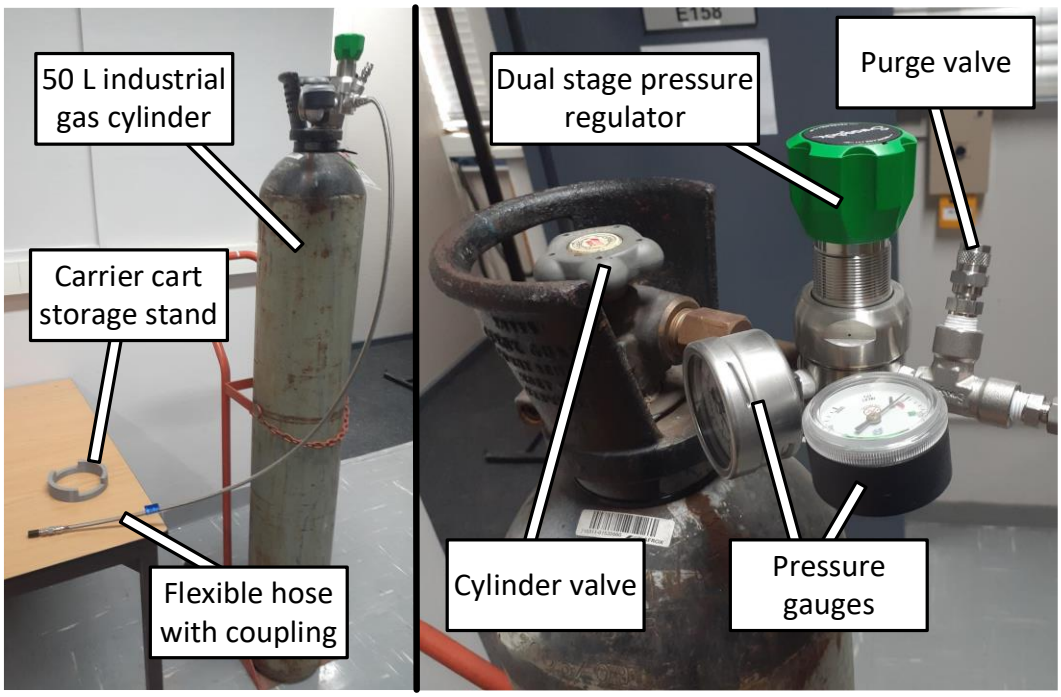


Figure 4.13 - Two annotated photographs showing the assembly of the propellant refill station.

4.6 Demonstration payload hardware design

In order to demonstrate the functioning of the satellite test facility as a whole, a demonstration payload was designed and implemented. This section looks at the hardware design of the demonstration payload, while Chapter 7 discusses the design of the cart control system implemented on this payload, and Chapter 8 looks at the practical results of the demonstration payload. The aim of the demonstration payload is to validate the functioning of the test facility by bridging

the feedback and communication system and the carrier cart to form a simple closed loop control system.

A block diagram showing the architecture of the demonstration payload is shown in Figure 4.14. The demonstration payload receives electrical power and controls the thruster array through the carrier cart's payload interface. The demonstration payload also communicates wirelessly with the test facility's feedback and communication system, receiving pose information and providing it with control output data.

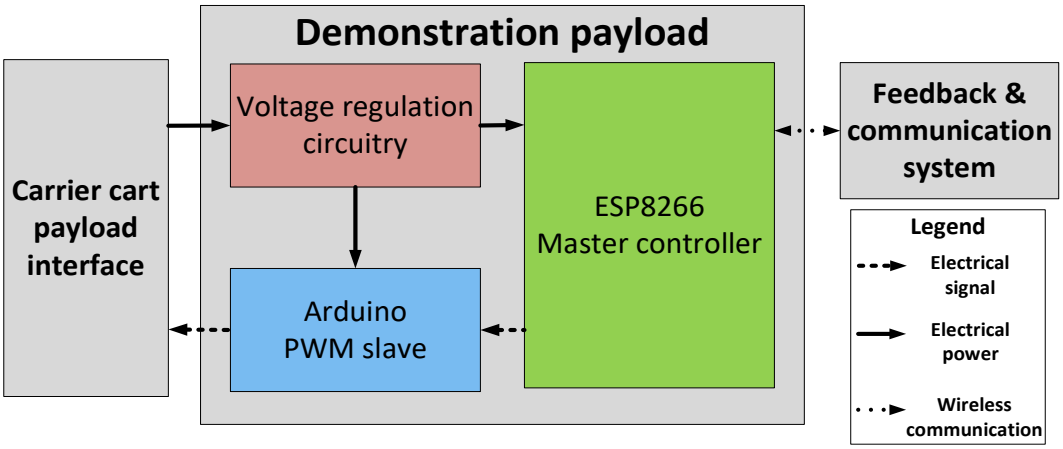


Figure 4.14 - Block diagram describing the architecture of the demonstration payload.

An ESP8266 development board is used as the master controller, which is responsible for bi-directional communication with the feedback and communication subsystem and for calculating control outputs to manoeuvre the cart. The chosen wireless communication technology is Wi-Fi, and the ESP8266 was chosen for its built-in Wi-Fi capabilities. The ESP8266 is programmed as the cart controller, which is discussed in detail in Chapter 7. It wirelessly receives reference signals and pose information for the cart, and accordingly calculates the control outputs to move the cart as desired. These control outputs are also sent to the feedback and communication system for logging purposes.

The demonstration payload makes use of pulse width modulation (PWM) to control the carrier cart's thruster array. This is done to approximate continuous control of the thrusters, thus simplifying the controller. The Arduino pulse width modulation (PWM) slave is necessary because the ESP8266's PWM is not readily capable of a frequency as low as 100 Hz, which is the frequency required to control the thruster array using PWM signals. The selection of 100 Hz as the PWM frequency is detailed in Chapter 7. The ESP8266 communicates the desired PWM duty cycles to the Arduino slave through a universal asynchronous receiver/transmitter (UART) connection. A baud rate of 500 kbaud is used to

ensure that the information is sent fast enough for the Arduino to process before adjusting its outputs. The PWM outputs of the Arduino are sent to the thruster array through the payload interface. The Arduino board used is an Arduino Pro Mini.

The ESP8266 and Arduino receive power from the carrier cart's battery pack through the payload interface. This electrical power is conditioned by some regulator circuitry before being supplied to the microcontrollers. The battery voltage is regulated down to 5 V, which is suitable to power both microcontrollers.

4.7 Summary and conclusion

This chapter detailed the design of the hardware-based subsystems of the test facility. The discussion was focused on the carrier cart subsystem, as it is the most hardware-intensive subsystem, but the propellant refill subsystem and demonstration payload were also looked at.

The carrier cart's main requirements were confirmed to be met by the design, aside from the runtime requirement - which necessitates that the thruster nozzles have a specific impulse of 25 s or higher at a thrust level of 0.4 N. The thruster nozzles were found to be difficult to machine locally and commercial solutions had very high costs. This brought upon the decision to investigate the feasibility and performance of 3D printed thruster nozzles. The following chapter details this investigation and the design & characterisation of the 3D printed nozzle solution.

5 Design & characterisation of 3D printed cold gas thruster nozzles

As discussed in the previous chapter, the procurement of nozzles for the cold gas thrusters (CGTs) used on the cart proved difficult. The department's in-house workshop was unable to machine the small geometrical profiles and features required - and the technical staff highlighted that it would be difficult to do so locally without very specialised and expensive tooling. It was therefore decided to investigate the feasibility and performance of 3D printed nozzles, as this would be a convenient and low-cost way to manufacture the nozzles. The customisability of these 3D printed nozzles also adds to the versatility of the cart, as nozzles with different characteristics can easily be designed and printed.

5.1 Desired specifications

From the requirements of the carrier cart and the interfaces of the thruster subassembly, the nozzle requires the following specifications:

- A thrust output of 0.4 N at a chamber pressure of 4 bar. Furthermore, to add to the versatility of the cart, it is desirable to have the option of 3D printing nozzles with thrusts ranging from 0.25 N to 0.75 N.
- A specific impulse of 24 s or higher at a working pressure of 4 bar to achieve the required runtime of the carrier cart. Unless otherwise indicated, stated pressure values refer to gauge pressure.
- An M5 threaded inlet port, with a flat surface for interfacing with a sealing gasket or O-ring.
- Rated chamber pressures of up to 6.7 bar to prevent failure in the case of human error when setting the carrier cart's pressure regulator.

5.2 Nozzle design

The theory and first principles used to design the CGT nozzles include the compressible flow theory outlined by Çengel & Cimbala (2014) and the CGT theory presented by Brown (2002) and Micci & Ketsdever (2000). The first consideration in the design was whether to use a simple converging profile or a converging-diverging profile. A converging-diverging profile was chosen for its increased specific impulse when compared to a converging profile. Specific impulse is a property of rocket engines which is related to the efficiency of the rocket in terms of propellant usage. Specific impulse is defined as

$$I_{sp} = \frac{F_{thrust}}{\dot{m}_N \cdot \mathrm{g}}, \tag{5.1}$$

where F_{thrust} is the rocket nozzle's output thrust, $\dot{m}_N$ is the mass flow rate of propellant through the nozzle, and g is the local acceleration due to gravity (Brown, 2002). A higher specific impulse is desired, as this means that a rocket is using less propellant to output a given force.

With nitrogen as a propellant, which is the case for the carrier cart, a theoretical specific impulse of 76 s is achievable in vacuum conditions (Brown, 2002). However, in atmospheric conditions the achievable specific impulse will be much lower. Furthermore, according to Brown (2002), for small CGT nozzles only about 90% of the theoretical specific impulse is typically achieved in practice.

Assuming fully isentropic, one-dimensional flow, and that supersonic flow occurs in the diverging section of the nozzle, the following equations hold (Brown, 2002; Çengel & Cimbala, 2014). The thrust coefficient is calculated as

$$C_f = \sqrt{\frac{2k^2}{k-1}\left(\frac{2}{k+1}\right)^{\frac{k+1}{k-1}}\left[1-\left(\frac{P_e}{P_c}\right)^{\frac{k-1}{k}}\right]} + \left(\frac{P_e-P_a}{P_c}\right)\frac{A_e}{A_t}, \tag{5.2}$$

where k is the specific heat ratio of the gas propellant, P_e is the exit pressure of the gas at the nozzle outlet, P_c is the chamber pressure at the nozzle inlet, P_a is the back pressure (i.e. the local ambient pressure), A_e is the exit area of the nozzle's profile, and A_t is the nozzle's throat area. This thrust coefficient can be used to determine the nozzle's output thrust by

$$F_{thrust} = C_f P_c A_t. \tag{5.3}$$

Finally, the mass flow rate can be calculated by

$$\dot{m}_N = \frac{P_c A_t}{\sqrt{RT_c}}\sqrt{k\left(\frac{2}{k+1}\right)^{\frac{k+1}{k-1}}}, \tag{5.4}$$

where R is the specific gas constant of the propellant and T_c is the stagnation temperature of the gas in the chamber.

Çengel & Cimbala (2014) have published more detailed compressible flow equations, which can be used to determine flow properties at various locations of the nozzle profile during operation. These more detailed equations were used for the design the features of the nozzle profile, but Equations (5.1) through (5.4) are sufficient for understanding the operation of the nozzle and how flow parameters and geometry affect the performance. The nozzle will be designed for optimal expansion, which implies that the exit pressure equals the back pressure. This is done to avoid shockwaves forming in the flow, which negatively affect the

performance. The effect of friction is negligible for short nozzles (Çengel & Cimbala, 2014), and is not included in Equations (5.1) through (5.4).

A noteworthy observation from Equation (5.2) is that the back pressure decreases the thrust output of the nozzle. For this reason, a given nozzle will always have a lower specific impulse in atmospheric conditions compared to vacuum conditions where the back pressure is effectively zero.

A sectioned sketch of the designed converging-diverging nozzle is shown in Figure 5.1. The nozzle was designed with the limitations of fused deposition modelling (FDM) 3D printing in mind. The design features an M5 inlet port for compatibility with the chosen solenoid valve, as well as a hexagonal outer perimeter around the end of the chamber to facilitate tightening the joint during assembly. The flat face perpendicular to the M5 port enables the use of an O-ring to from a seal between the nozzle and the solenoid valve. As deduced from Equations (5.1) through (5.4), the geometrical parameters which are key to the nozzle's thrust output and performance are the throat diameter and the exit diameter.

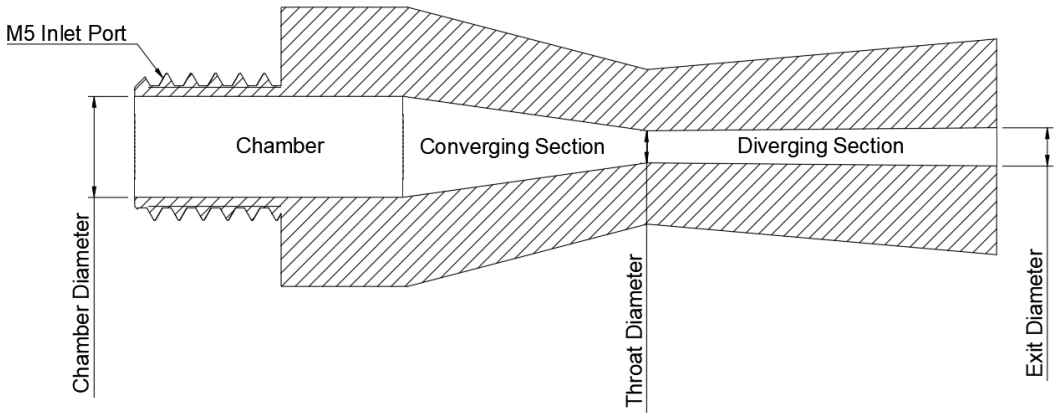


Figure 5.1 - A sectioned drawing of the designed converging-diverging CGT nozzle.

For the case of an ideal converging-diverging nozzle used in atmospheric conditions on Earth, with fixed fluid properties, chamber pressure, and back pressure (which is the case for the nozzles designed for the carrier cart) – the nozzle's specific impulse only depends on the ratio of the exit diameter and the throat diameter. Thus, for a nozzle with a given pressure ratio between its chamber pressure and the back pressure, an optimal diameter ratio exists which produces the highest specific impulse for that nozzle's conditions. The relation between the diameter ratio and the specific impulse is plotted in Figure 5.2. The plot assumes a converging-diverging nozzle with a fixed chamber pressure, back pressure, and fluid properties. The ratio of the chamber pressure to the back pressure in the plot is 5, which is the case for the carrier cart's nozzles. From this

plot it is deduced that the optimal diameter ratio for the case of the carrier cart's nozzles is 1.16, which yields a theoretical specific impulse of 48.8 s. This is the diameter ratio used for the nozzles. At this diameter ratio, the nozzle is designed for optimal expansion, which means that the exit pressure is equal to the back pressure and neither subsonic flow nor shockwaves negatively affect the performance.

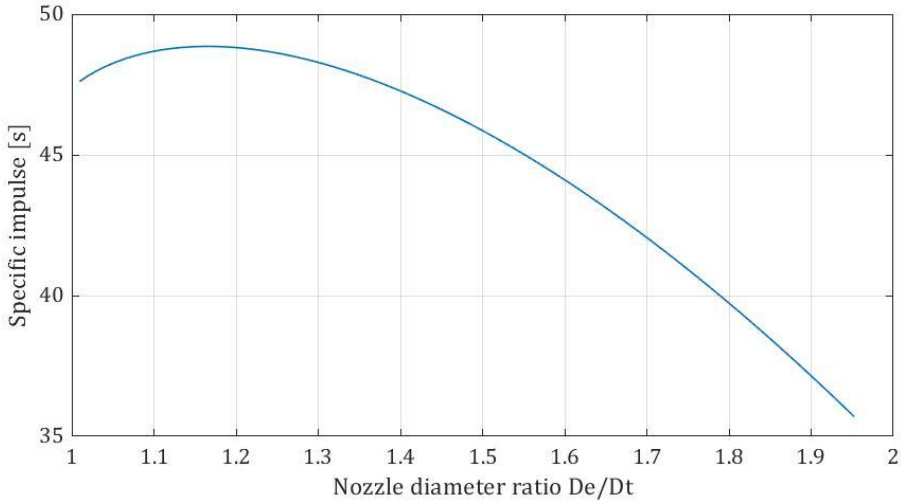


Figure 5.2 - A plot relating the nozzle diameter ratio to achieved specific impulse for the case of the nozzles designed for the carrier cart.

The nozzle thrust output desired for the carrier cart is 0.4 N, but nozzles with output thrusts in a range of 0.25 N to 0.75 N will be characterised to investigate the viability of 3D printing nozzles in this range. Four nozzle groups will be tested with thrusts of 0.25 N, 0.4 N, 0.5 N and 0.75 N. The designed dimensions of the throat and outlet of each nozzle are listed in Table 5.1. The theoretical specific impulse for each nozzle is the same at 48.8 s.

Table 5.1 - Throat and exit diameters of the four groups of nozzles designed for different output thrusts.

Thrust output [mN]	Throat diameter [mm]	Exit diameter [mm]
250	0.74	0.86
400	0.95	1.10
500	1.05	1.22
750	1.29	1.50

As mentioned before, for the ideal case of one-dimensional flow, only the throat diameter and exit diameter of the nozzle affect the flow through the nozzle. However, as highlighted by Hamedi-Estakhrsar et al. (2018), in practice the converging and diverging angles also play a role by introducing so-called angularity losses. Using large converging and diverging angles reduce the validity of the one-dimensional flow assumption made during the design and more inaccuracies will be introduced. With the results found by Hamedi-Estakhrsar et al. (2018) in mind, a converging angle of 16° and a diverging angle of 1° is used for the designed nozzles. These values minimise angularity losses. Furthermore, the small angles chosen will reduce the surface roughness of the internal profile of the 3D printed nozzles, which is desired because the surface roughness introduced by the FDM 3D printing process is expected to cause frictional losses.

The chamber diameter is 2.5 mm to facilitate relatively low inlet speeds, since the design assumes low Mach numbers for the flow speed at the inlet. The minimum wall thickness of the nozzle is 2 mm to allow it to withstand chamber pressures up to 6.7 bar.

5.3 Feasibility of 3D printed CGT nozzles

Fused deposition modelling (FDM) is the chosen 3D printing process for manufacturing the nozzles, as recent innovations have made it widely accessible and even low-cost machines can produce functional parts. However, the proposed solution of 3D printing the carrier cart's CGT nozzles using FDM technologies brings with it its own set of challenges.

The first uncertainty was whether a low-cost desktop 3D printer could successfully print the small features required. The 3D printer to be used for manufacturing the nozzles is the stock Creality Ender 3, which is a low-cost entry-level 3D printer. The stock Ender 3 uses a 0.4 mm hotend nozzle for printing. Polylactic acid (PLA) is used as a printing material because of its good printability and sufficient mechanical strength.

The design calculations showed that the throat diameter of the desired nozzles would be on the order of 0.75 mm and larger, so it was investigated whether the printer could successfully print this hole size. The problem with FDM 3D printing such small holes is that they tend to fuse shut because of the cohesiveness of the molten plastic. To investigate the feasibility of printing such small holes, a few prototypes with a throat diameter of 0.75 mm were printed. Figure 5.3 shows one of the early prototypes as it was printed on the print bed. All the prototypes' profiles were successfully printed, and no fusing occurred at the throat of the nozzle.

The next challenge was to print the M5 male port of the thruster. This introduced a practical constraint where the nozzle now had to be printed with its outlet face on the print bed. Note that in contrast, the prototype shown in Figure 5.3 is printed on its inlet face. The issue with this is that the first layer tends to be more compressed than other layers, and so the exit diameter was fusing shut because of its small size. To solve this issue, the print bed was re-levelled using precision feeler gauges, which reduced the compression of the first layer. With the more precisely levelled print bed, the printer was seen to produce accurate dimensions on the first layer and the fusing problem was eliminated.

Figure 5.3 - An early nozzle prototype printed to determine the feasibility of printing holes as small as 1 mm.

The Ender 3 printer was found to be capable of printing the nozzles without any fusing of holes occurring, and could successfully print the M5 port required for interfacing with the solenoid valve. Figure 5.4 shows some of the 3D printed nozzles making use of the final design – these nozzles are from the 0.25 N, 0.5 N, and 0.75 N groups. Now that the nozzles were successfully manufactured, the next step was to characterise them and evaluate their performance.

Figure 5.4 - A photograph showing some of the printed nozzles to be characterised, with a coin for scale.

5.4 Experimental test rig design

This section discusses the design of the experimental test rig used to characterise the 3D printed CGT nozzles. The test rig is designed to measure parameters which are key to the evaluating performance of the nozzles – which are the output thrust, the volumetric flow rate of the propellant through the nozzle, the chamber pressure and finally the temperature of the gas flowing into the chamber. The test rig was designed with the assistance of the guidelines for mechanical measurements as outlined by Figliola & Beasley (2015).

5.4.1 Hardware design

Figure 5.5 shows a photograph of the constructed experimental test rig. The test rig receives electrical power from a DC power supply , and receives nitrogen gas at a regulated pressure from the propellant refill station discussed in Section 4.5. The test rig outputs measured data (containing force and temperature measurements) via universal serial bus (USB) communication to a receiving device such as a computer or smartphone.

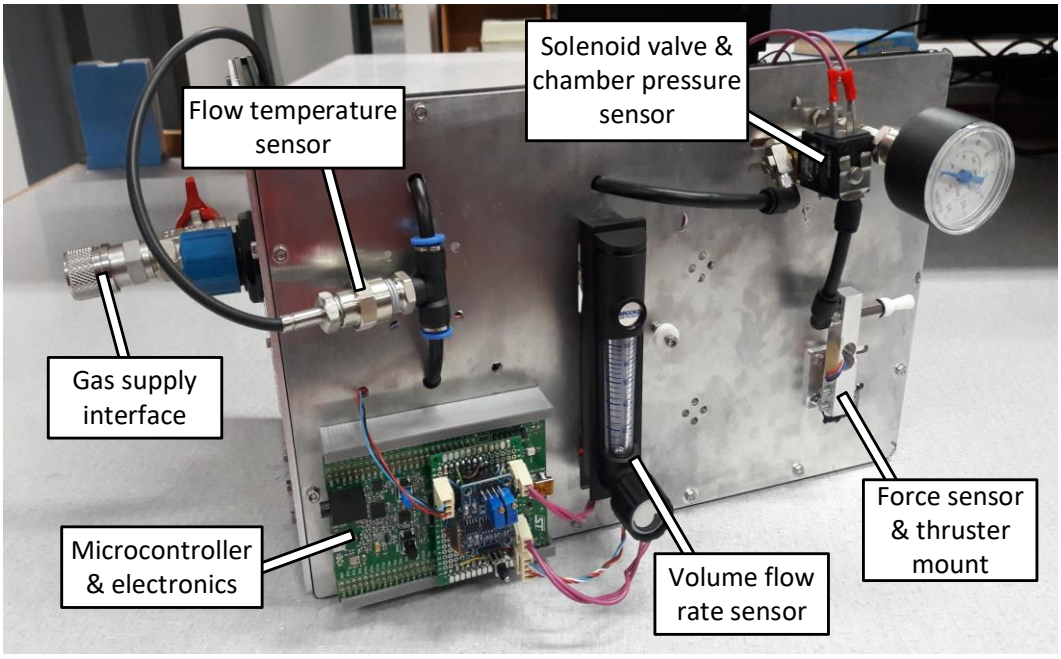


Figure 5.5 - A photograph of the constructed experimental test rig.

The gas supply interface connects the test rig to the propellant refill station. The gas supply interface consists of a quick coupling for connection to the refill station, a ball valve for turning the supply on and off, and a relief valve to protect the test rig's components from damage. The propellant refill station is used to supply nitrogen at a regulated pressure of 4 bar to the chamber of the nozzle.

The nitrogen flows from the gas supply interface through a fixture with a negative temperature coefficient (NTC) probe, which measures the temperature of the gas flowing to the nozzle's chamber. Downstream of this, the nitrogen flows through a variable area flowmeter to measure the volumetric flow rate through the nozzle. After this, the gas flows to the solenoid valve inlet, which controls the flow through the nozzle. A pressure gauge is used to measure the chamber pressure of the nozzle during operation. This gauge is also used to set the chamber pressure to 4 bar during testing.

A microcontroller is used to control the solenoid valve during the experiment and to sample the electrical sensors used. The test rig also uses auxiliary electronic circuits to condition the signals output from the sensors, and to condition the output signal used to control the solenoid valve. The resistance of the NTC probe changes according to the measured temperature of the flow, and this change in resistance is measured using a voltage divider circuit. A load cell is used to measure the force output of the nozzle, and its strain gauge is connected in a wheatstone bridge configuration. The output voltages of the bridge are very low, on the order of 2 mV/N, and so an instrumental amplifier is used to amplify its output to a level which the microcontroller can sample using its analogue-to-digital converter (ADC). A driver circuit identical to the one used on the carrier cart, as described in 4.4.4, is used to condition the digital output signal from the microcontroller to a 24 V signal to control the solenoid valve. As mentioned before, a DC bench power supply is used to supply power to the microcontroller, the electrical circuitry, and the solenoid valve.

5.4.2 Software design

The microcontroller was programmed to control the solenoid, sample the electrical sensors, and to output the measured data over USB during the testing. Two different testing modes were programmed, and the mode can be changed using a toggle switch.

The first mode simply allows the user to press a button to turn the solenoid on and off. In this mode, the microcontroller samples and outputs the data at 2 Hz. The intent of this mode is to allow the user to read the volumetric flow rate and the chamber pressure from the test rig's mechanical sensors. Furthermore, the user can read the nozzle's thrust output and the flow temperature from the USB log.

The second mode initiates the testing by determining a dynamic offset – which is used in the calculations. This dynamic offset is used with the load cell calibration curve to minimise the effects of sensor hysteresis, which is an infamous problem with load cells (Figliola & Beasley, 2015). The dynamic offset is calculated by activating the thruster for a few seconds, then deactivating it for a few seconds and measuring the thrust level when there is no flow through the nozzle. The measurement is then effectively zeroed. After this initial sequence, the mode

allows the user to send a control signal to the thruster which has a square waveform. This square waveform has a 1 second period and a 50% duty cycle and is output for 10 cycles (equating to 10 s). When this second mode is active, the microcontroller outputs over USB the sampled force output, flow temperature, log time, and the state of the control signal. This data is sampled and transmitted at a frequency of 100 Hz. The square waveform used to control the thruster allows for a more accurate measurement of the average thrust output, but the data requires some post-processing.

5.4.3 Sensor calibration

The NTC probe used for measuring temperature is factory calibrated, and its output characteristics are listed in its datasheet. The NTC probe's resistance has a non-linear relationship to the measured temperature, and so the output characteristics in the datasheet were used to construct a lookup table in the microcontroller software. The microcontroller program measures the resistance of the probe and uses the lookup table to determine the corresponding temperature. A pre-calibrated thermocouple was used to verify the implementation of the lookup table, and the two sensors produced results within 1°C of each other, which is acceptable for this application.

The test rig's variable area flowmeter is also factory-calibrated, with a specified accuracy of 4%. No further in-situ calibration was performed.

The load cell was calibrated using a set of precision masses. The load cell's amplified output was measured for masses ranging from 0 kg to 0.2 kg, which corresponds to the force range of 0 N to 2 N. This data was fitted to a curve using linear regression, and the resulting parameters are used by the microcontroller program for calculating the force measured by the load cell.

5.5 Characterisation results

This section discusses the results from characterising the solenoid valve and each group of nozzles. There are four groups of nozzles – one for each designed thrust output level – and each nozzle group consists of 5 nozzles 3D printed with the same gcode. The gcode is the source file which the FDM printer uses to created the 3D printed object. Furthermore, the four groups of nozzles were initially printed using the baseline slicer settings and characterised – this is discussed in Subsection 5.5.2. After this characterisation, some tuning was done in the slicer software and the four groups were printed again with the tuned settings to achieve better nozzle performance – these results are discussed in Subsection 5.5.3. The slicer is the software used to convert the computer-aided design (CAD) model to gcode, and contains many parameters for tuning 3D prints.

5.5.1 Solenoid opening/closing time

The solenoid opening/closing time was measured with the assistance of the experimental test rig. This measurement is important for the purposes of modelling the carrier cart for the control system simulations in Chapter 7. The opening time measurement was done by using an oscilloscope to find the time difference between the point in time where the control signal is pulled high, and the time at which the load cell measurement reached 90% of its final measured value. The closing time was measured similarly. Both the opening time and the closing time were measured as 9 ms.

5.5.2 Results with baseline slicer settings

The baseline slicer settings used are the default settings for the Ender 3 printer in the Cura 4.2.1 slicer software, with some small exceptions that were made to yield better print quality in the initial feasibility investigation. The exceptions are:

- A 0.12 mm layer height.
- A print speed of 30 mm/s.
- Combing mode set to 'Not in skin'.

An example dataset for one of the characterised nozzles is shown in Figure 5.6. This measurement was taken using the second mode of the test rig's microcontroller program as described in Subsection 5.4.2. The nozzle linked to this set of measurements was designed for an output thrust of 500 mN and was printed using baseline settings.

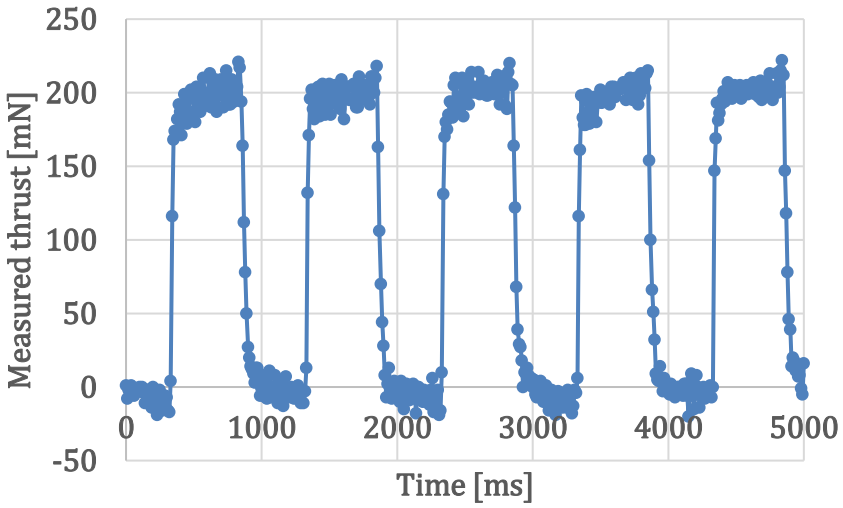


Figure 5.6 - An example dataset for a single baseline nozzle, showing the thrust measurement versus time.

The results for the four groups of baseline nozzles are shown in Table 5.2. Not shown are the measurements of the volumetric flow rate and the flow temperature, as the are used to determine the specific impulse. Recalling from Section 5.2, the theoretical specific impulse for all the nozzles is 48.8 s.

Table 5.2 - Nozzle characterisation results for nozzle groups printed using baseline settings.

Designed thrust [mN]	Average achieved thrust [mN]	Thrust standard deviation [mN]	Average achieved specific impulse [s]
250	46	13	43.5
400	161	21	32.4
500	208	26	29.3
750	318	29	25.6

As gathered from the results, none of the nozzles printed using baseline settings achieved the designed output thrust level. The lower force output is speculated to be caused by the throat diameters being printed smaller than they are designed. This occurs because of the flaws of the FDM manufacturing process, which tends to decrease the size of printed holes compared to the CAD model. Fortunately, the slicer software (or even the CAD software) can be used to tune the throat diameters until the thrust outputs are at the desired values.

The nozzle designed for 250 mN output thrust obtained a measured specific impulse of 43.5 s, which is 89% of it's theoretical value. As mentioned before, the maximum specific impulse typically achieved by small nozzles in practice is 90% of their theoretical specific impulse (Brown, 2002) – the nozzle designed for 250 mN achieves this. However, its thrust output is only 20% of what it was designed for. The other three groups of nozzles achieved specific impulses significantly lower than their theoretical values. This is a problem because it means that significant losses are occurring in these nozzles. Furthermore, specific impulse is seen to decrease with increasing thrust level (and thus increased flow rate through the nozzle). This leads to the speculation that the losses are frictional losses. The low specific impulses achieved will be investigated further in Subsection 5.5.4.

Each group of nozzles achieved a reasonably low standard deviation of thrust outputs. This implies that the nozzles printed from the same gcode are quite repeatable.

5.5.3 Results with tuned slicer settings

Because of the very small diameters being printed, the limitations of FDM technology are causing the throat diameters to be printed smaller than intended. Since the force output of the nozzles are very sensitive to their throat diameters, this problem has a significant effect on the nozzle performance. Fortunately, most slicer software packages have parameters that can be tuned to achieve more accurate holes. For the case of the Cura slicer used, the parameter of interest is the 'Horizontal expansion' setting. According to the description in the software, this parameter applies an offset to all polygons in each layer. This allows the user to effectively increase the size of a hole by entering a negative value. The offset is applied to the inner faces of each polygon, and so setting the parameter to a value of -0.1 mm effectively increases vertical hole diameters by 0.2 mm. This parameter was tuned individually for each group until the measured nozzle output thrusts were within 10% of the designed values.

Table 5.3 shows the results of the characterisation of the four groups of tuned nozzles. For the case of the tuned nozzles, three out of four groups successfully achieved an output thrust within 10% of their designed value. For the case of the nozzle designed for 750 mN of thrust, since the flow rates are higher than expected for the nozzles, the test rig (and similarly the carrier cart) is not capable of maintaining a chamber pressure of 4 bar. This effectively sets a maximum thrust limit of 670 mN on the carrier cart when these 3D printed nozzles are used.

Table 5.3 - Nozzle characterisation results for nozzle groups printed using tuned settings.

Designed thrust [mN]	Average achieved thrust [mN]	Thrust standard deviation [mN]	Average achieved specific impulse [s]	Horizontal expansion setting [mm]
250	241	29	28.7	-0.15
400	405	24	27.5	-0.18
500	465	22	27.0	-0.2
750	670	32	22.3	-0.3

The tuned nozzle groups once again showed a relatively low standard deviation and also had significantly lower specific impulses than designed for. The low specific impulses will be investigated in the following subsection.

For the case of the nozzle designed for 400 mN output thrust - which will be used on the practical implementation of the carrier cart - even though its specific impulse is lower than its theoretical value, it still meets the required specification of 24 s or larger. Furthermore, the thrust specification is met within 10% of the required value, and all the nozzles were able to withstand 6.7 bar chamber pressure. Thus, all the required specifications are met by the nozzle.

5.5.4 Investigation of low specific impulse

The low specific impulses seen during the characterisation are speculated to be caused by either propellant leakage through the slightly porous walls of the nozzle or by frictional effects which were not incorporated in the design. To investigate this, two tests were conducted.

The first test involved printing a nozzle with a blocked exit to determine how much leakage was occurring through the walls of the nozzle. The blocked nozzle was installed on the test rig and the chamber was pressurised to 4 bar. The leakage through the walls was found to be negligible in this test, and so the porosity of the 3D printed walls does not seem to play a role in the low specific impulses observed.

The second test involved printing a nozzle with a longer flow path to investigate the effects of friction in the nozzle. Friction was suspected to be causing efficiency losses because of the roughness of the 3D printed inner walls of the nozzle, as well as the fact that the specific impulse was seen to decrease with increasing flow rate. The friction in the nozzle is also likely causing the gas to over-expand in the nozzle's diverging section, which in turn leads to an output force reduction because the exit pressure is lower than the back pressure (see Equation 5.2). Furthermore, this means that the nozzle is not achieving optimal expansion, lowering the specific impulse.

The nozzle with a longer flow path was printed by halving the diverging angle, doubling the length of the diverging section. The longer flow path will cause the gas to face more friction and thus the frictional losses will be increased. This longer nozzle was installed and characterised on the test rig. It was found that both the force output and the specific impulse for the long nozzle were significantly lower than its shorter counterpart. This confirms that friction is the main cause of the losses seen in the nozzles 3D printed using FDM technology. For this reason, it is inferred that shorter nozzles will perform better and are thus preferred.

The surface roughness could possibly be improved by using other 3D printing technologies. Stereolithography and selective laser sintering typically produce better surface finishes than FDM printers, and these technologies could be looked at for manufacturing nozzles in the future. However, for the purposes of the test facility, the carrier cart's requirements are met by the tuned nozzle which was designed for 400 mN of output thrust.

5.6 Parameterised nozzle model

To add further versatility to the test facility, the CAD model of the designed nozzle was parameterised. This parameterisation means that future users can enter a desired output thrust value and the CAD software will generate the required dimensions and apply them to the model automatically. The model can then be exported for 3D printing. This allows an abstraction layer to be created for future users, so that they do not need to be familiar with the design of nozzles to use a nozzle with a different thrust level on the carrier cart.

The trends seen for the horizontal expansion settings during the tuning of the nozzles were also applied to the parameterised model. This was done to try and minimise the amount of tuning that future users need to do. However, it is still recommended to use the experimental test rig to confirm the specifications of the objects produced by the parameterised model.

5.7 Summary and conclusion

This chapter discussed the design and characterisation of the 3D printed cold gas thruster nozzles used on the test facility's carrier cart. The nozzles were designed to be printed using fused deposition modelling technology and were designed to interface with the solenoid valves on the carrier cart. An experimental test rig was designed to characterise the 3D printed nozzles and determine how well they perform. Most of the nozzles were measured to have significantly lower specific impulses compared to their designed values. It was found that the lowered specific impulses are caused by frictional losses, which are significant because of the roughness of the internal walls formed by the 3D printed process.

The 0.4 N nozzle designed for use by the carrier cart was seen to meet all specifications required by the cart. It achieved a specific impulse of 27.5 s, which is higher than the required 24 s. The carrier cart is thus ready for integration into the test facility. The next chapter looks at the design of the feedback and communication system, which will be used to estimate the pose of the carrier carts on the table and relay this data to the user.

6 Feedback and communication system design

This chapter discusses the design of the feedback and communication system (FCS) – which, among other functions, provides the user with a means of interacting with their experiment. The FCS is responsible for measuring, displaying, and logging pose information (and other data) for the experiments done on the test facility. Furthermore, the FCS is designed to wirelessly communicate with the carrier cart's payload if desired.

6.1 System requirements

The feedback and communication system should be capable of the following:

- Determine the planar translational and rotational position of the carrier carts, relative to a fixed point on the glass platform, to an accuracy of within 0.03 m and 0.1 rad respectively.
- Determine the planar translational and rotational velocity of the carrier carts, relative to a fixed point, with a noise amplitude of less than 0.05 m/s and 0.1 rad/s respectively, when stationary.
- Facilitate bi-directional wireless data transfer with the payloads of the carrier carts, allowing pose information to be sent to the payloads and allowing control information to be received from the payloads.
- Provide a user interface with which the user can observe and log pose information, as well as send and receive data to and from their payloads if desired.
- Achieve the above for the case of multiple carrier carts on the glass platform.
- Output measured pose information at a rate of 10 Hz or lower.

6.2 System architecture

Figure 6.1 shows the architecture of the FCS. The system consists of three main subsystems – the communication manager, the optical pose estimator, and the user interface.

The communication manager is responsible for wirelessly sending and receiving data to and from the carrier cart, as well as internal data transfer between

software executables. The communication manager essentially acts as a data relay connecting the optical pose estimator, the user interface, and the payloads, allowing them all to communicate with one another. Furthermore, it manages and facilitates the communication between software components internal to the user interface and the optical pose estimator.

The optical pose estimator's role is to determine the pose of each carrier cart on the table relative to a fixed reference point. The pose estimator is also capable of logging this information and displaying it in near real-time on the user interface. The optical pose estimator consists of a single camera, a set of fiducial markers, and a pose estimation algorithm implemented in software.

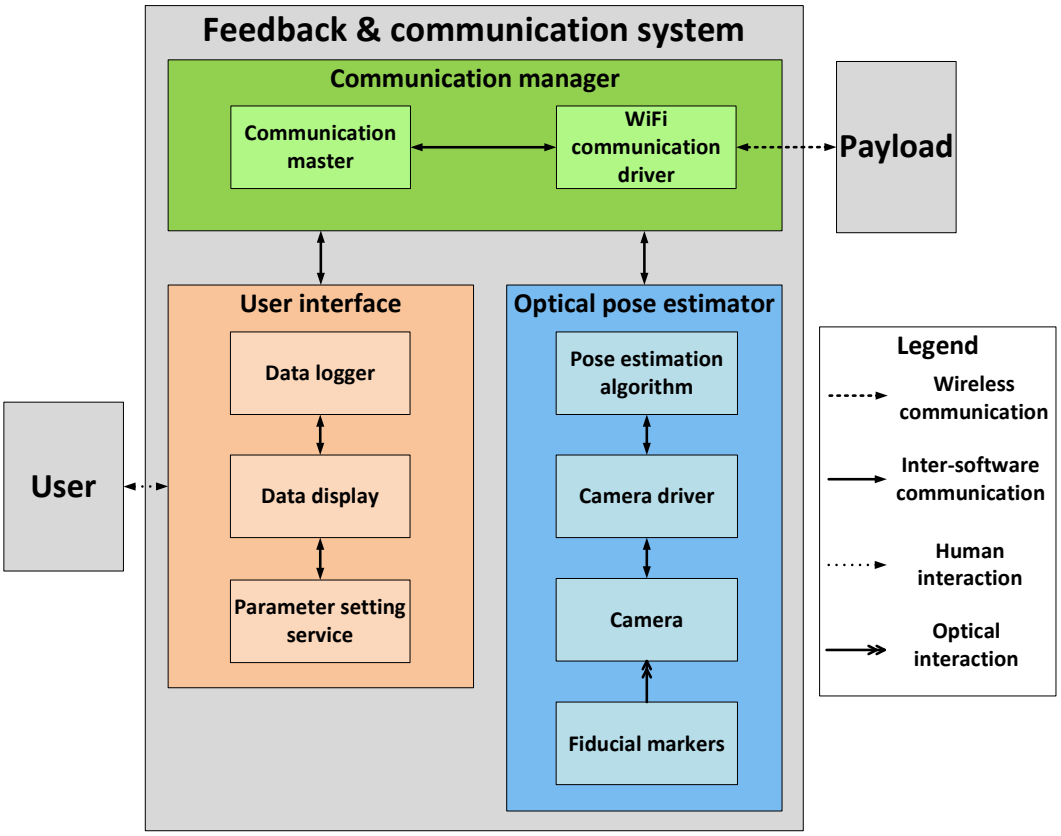


Figure 6.1 - A block diagram showing the architecture of the feedback and communication system.

The user interface enables interaction with the active experiment. It allows the viewing and logging of current pose information and the sending/receiving of data to/from the payloads. This allows, for example, for control parameters to be changed during an experiment and for the user to view and log control outputs as communicated by their payloads.

6.3 Overview of the ROS framework

6.3.1 Introduction to the ROS framework

The implementation of the software components of the FCS makes use of the open-source middleware named Robot Operating System (ROS) (Robot Operating System, 2010). ROS is a set of tools and libraries, typically run on a Linux system, intended to streamline the design of robotics control and communication software. Fundamentally, ROS in its most basic form is a communication infrastructure – and so it is used to handle the necessary communication between the components of the FCS.

A ROS network typically consists of executable components referred to as nodes, which communicate with each other through a publish-subscribe model. For example, one node may publish specific messages to a topic name while another will subscribe (listen) to that topic, receiving any messages published to that topic. Every ROS network also has a ROS master node which, amongst other functions, assists in communications between all the nodes. A diagram describing the communication between ROS nodes is shown in Figure 6.2. The ROS master is responsible for registration and naming of nodes and topics, enabling the nodes to communicate with each other. Each node will typically have a single function, such as sending position commands to a robot or detecting a marker's position from raw images. A collection of ROS nodes must thus work collaboratively to perform a more complex task - such is the case of the FCS.

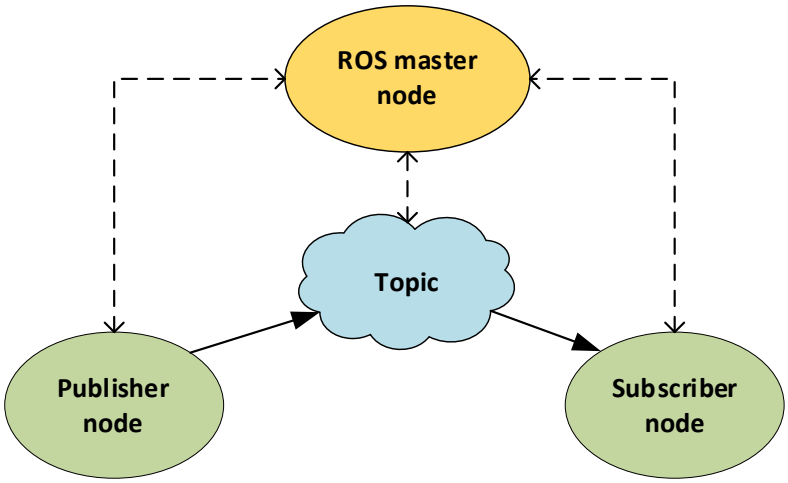


Figure 6.2 - A diagram describing the typical communication model of ROS.

A collection of ROS executables, scripts, configuration files, libraries and other software modules is called a ROS package. There are many open-source packages (written by the ROS community) available on the ROS package index. The wide

variety of packages available perform various functions to solve problems typically found in robotics applications. Some well-established ROS packages are used for certain elements of the FCS, while other elements consist of custom-written or adapted software using open-source libraries. Furthermore, the software elements of the FCS are organised into a custom-created ROS package to ease further development and additions by future users.

The messages communicated over ROS topics have specific types, and each message type is structured similarly to a class in object-oriented programming. Each topic expects a certain type of message, and only accepts that message type. A message consists of a list of named variables, which can have different variable types such as integers, floats, and strings - among other common types. Messages can also contain arrays of variables and can even nest other types of messages. Many standard ROS message types are available for common use cases and are organised into ROS packages. For the case of the FCS, custom ROS messages were created as part of the test facility's custom ROS package. The FCS uses a mixture of standard ROS messages and the custom messages created.

The choice of ROS as the framework and the use of open-source libraries for the system in general is attributed to the ease of integration and expansion that open-source software provides. It will also help future users to understand, use and expand the test facility's features for their own scenario more easily. The nature of ROS projects also opens the test facility to be used for general robotics experiments, along with its satellite-oriented experiments. This all adds to the versatility and expandability of the test facility.

6.3.2 Implementation of the feedback and communication system within the ROS framework

To facilitate communication between the optical pose estimator, the user interface, and the communication manager itself, all three subsystems' software components are built into the ROS framework as collections of ROS nodes and utilities. Figure 6.3 shows an overview of the software components that make up the FCS and indicates the groupings that form part of each subsystem. To avoid clutter, the ubiquitous connections between the ROS master node and all the other ROS elements are omitted in Figure 6.3.

Section 6.4 focuses on the design of the communication manager subsystem. Section 6.5 details the optical pose estimator subsystem's design, while Section 6.6 details the design of the user interface subsystem.

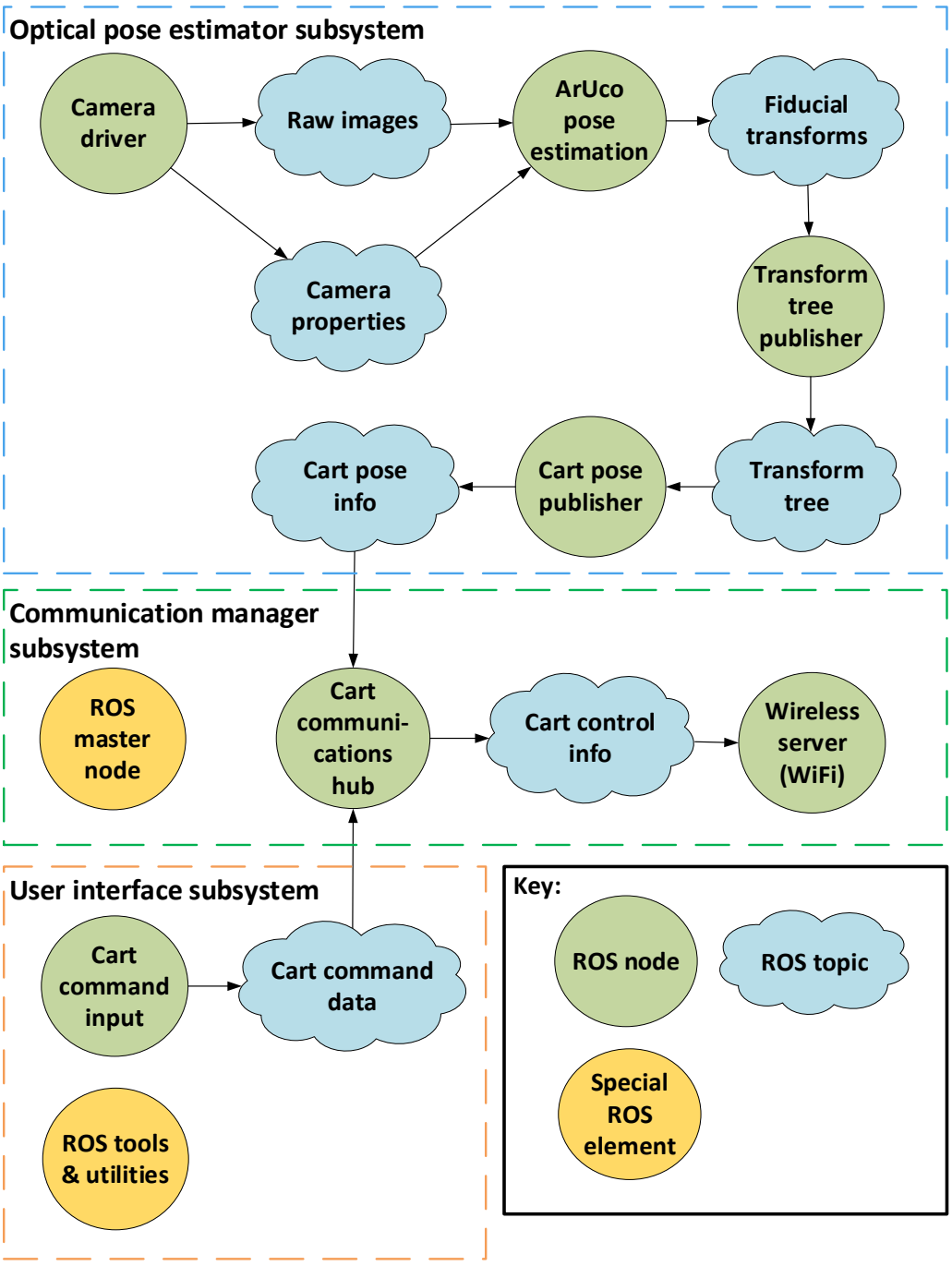


Figure 6.3 - An overview of the feedback and communication network's software components as implemented in ROS.

6.4 Communication manager design

The communication manager is responsible for enabling the communication between the software components of the FCS, as well as the communication

between the system and the payload on the carrier cart through WiFi. As described in Figure 6.3, the three main ROS nodes that make up the communication manager subsystem are the *Cart communications hub* node, the *Wireless server node*, and the ROS master node.

As mentioned previously, the ROS master node facilitates communication between all the ROS nodes in the system. This enables the communication between the software components (nodes) internal to the optical pose estimator and the user interface. It also enables the communication between the user interface and the optical pose estimator. This allows for capabilities such as setting pose estimation parameters and displaying pose information and other data in near real-time to the user.

The *Cart communications hub* node collects the most recent relevant data from the optical pose estimator and the user interface and relays it the to the *Wireless server* node. The *Cart communications hub* node is also responsible for controlling the rate at which data is sent to the cart. The messages are sent over the *Cart control info* topic contains an array of data indexed according to the carrier carts' ID numbers. The data includes the current pose information of each cart, and any relevant data which was input by the user. For an example of relevant user input data, the demonstration payload receives reference inputs and control parameters from the user interface subsystem.

The *Wireless server* node is responsible for subscribing to the *Cart control info* topic and wirelessly relaying the received messages to the carrier cart payloads. The *Wireless server* node does this by interfacing with the host computer's WiFi drivers and broadcasting the data over a local network. The carrier cart payloads connect to the ROS master over WiFi and receive the array of data periodically. Each payload must then access the data relevant to their respective carrier cart by indexing the received array according to their unique ID, which must be assigned by the user when launching the test facility's ROS application. The *Wireless server* node is capable of bi-directional communication and is able to receive and relay data from each carrier cart to the ROS master node. This allows relevant control data from each carrier cart to be available for the user to view and log. The rate of data transfer to the carrier cart can be set by the user changing a parameter at start-up. The *Wireless server* node makes use of the 'rosserial' ROS package (ROS Rosserial Documentation, 2018).

The bi-directional communication feature of the test facility has three different use cases:

1. Parts of the cart control system, such as the sensor feedback, are implemented external to the payload. In this case the pose data is sent to the payload and control info is sent back. This is the case for the demonstration payload.

2. The cart payloads operate without receiving feedback from the FCS (i.e. they make use of their own sensors), but they send control information to the FCS to be displayed to the user or logged.

3. The cart payloads operate completely independently of the FCS and there is no need for data exchange. In this case the user can simply use the data logged by the FCS and compare it to their own measurements.

6.5 Optical pose estimator design

6.5.1 Pose estimation strategy

The optical pose estimator is responsible for estimating and communicating the positions and velocities of all the carrier carts on the glass platform, relative to a reference marker. The optical pose estimator achieves this by use of a camera, a set of fiducial markers, and a pose estimation algorithm to approximate the pose of the carrier carts.

The fiducial markers used are the open-source ArUco markers, since they offer robust and fast detection, and are simple to work with. The chosen set of markers is the 6x6 ArUco dictionary, as this set allows a good balance of fast and robust detection of the ArUco markers. It is, however, easy to change this in the software and using a different dictionary only requires changing one parameter. This allows users the option to use other dictionaries if the experiment requires it.

Some examples of the ArUco markers used are shown in Figure 6.4. Each marker has a unique ID which is used by the pose estimating algorithm to distinguish between the reference marker and each individual carrier cart. One marker is used as a reference marker (also called the origin marker) and is mounted on the glass platform. The default location of this reference is the bottom left corner of the table as seen by the camera but can be moved by users if desired. Figure 6.5 and Figure 6.6 show the hardware solutions used for mounting of the ArUco markers on the glass platform and carrier cart respectively. The mounting hardware is designed such that the cart markers and the reference markers are at the same height to minimise projection errors in software. However, if an user's payload requires more height clearance the software can handle the markers being at different heights, likely at the cost of some accuracy. The markers used have a side length of 11 cm. The optical pose estimator can accommodate smaller markers, down to a practical limit of about 5 cm, but larger markers are used to improve reliability of detection.

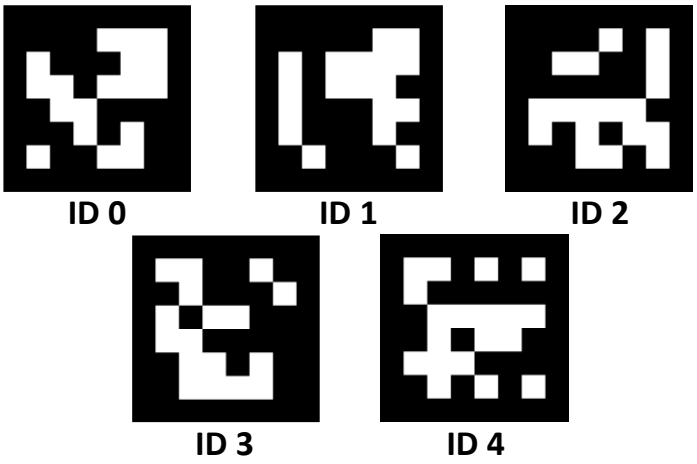


Figure 6.4 - Some examples of the 6x6 ArUco markers used for the pose estimation.

Figure 6.5 - Mounting hardware used for mounting the reference marker to the glass platform.

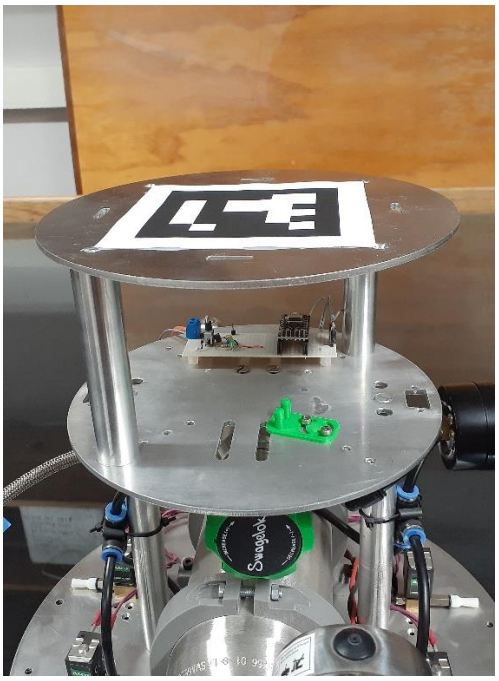


Figure 6.6 - Mounting hardware used for mounting the unique cart markers to each individual carrier cart.

6.5.2 Pose estimation algorithm and auxiliary software components

Recalling Figure 6.3, the optical pose estimator's software components are recapped in Figure 6.7. The pose estimation algorithm is executed in the *ArUco*

pose estimation ROS node. The *Transform tree publisher* and *Cart pose publisher* nodes play an auxiliary role in that they use the pose estimation data and transform it into the form desired for displaying, logging and sending it to the carrier cart payloads. The *Camera driver* node is responsible for interfacing with the physical camera and is discussed in Section 6.5.3 .

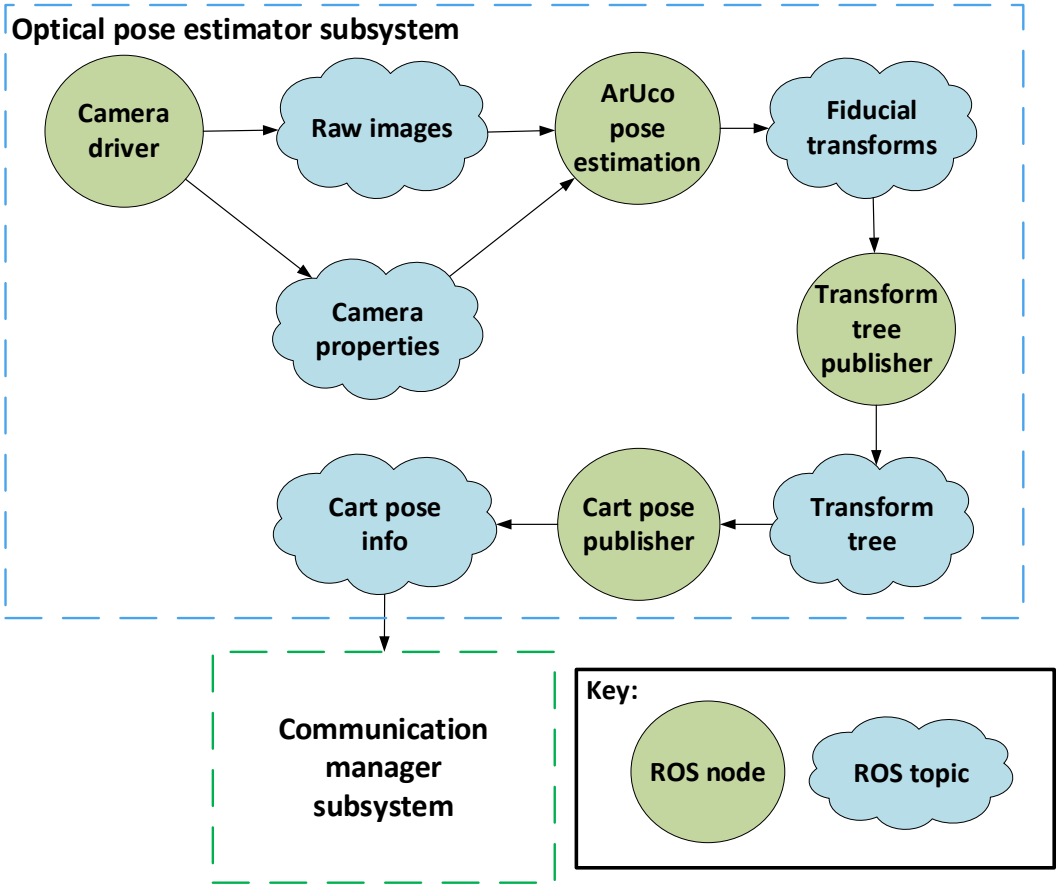


Figure 6.7 - A recap of the software elements of the optical pose estimator's software components, extracted from Figure 6.3.

The pose estimation algorithm makes use of the open-source computer vision library OpenCV, initiated by Bradski (2000). The key functions used form part of OpenCV's ArUco module developed by Garrido-Jurado et al. (2014). More detailed information about the ArUco module used can be found in the OpenCV documentation (The OpenCV Reference Manual, 2018).

The process that the algorithm follows to detect the ArUco markers and estimate their pose is described in Figure 6.8. The 'fiducials' ROS package (ROS Fiducials Documentation, 2018) is used for this node.

At the start of each cycle of the pose estimation algorithm, a raw image and a message containing the camera properties are received from the *Camera driver* node. These are the inputs required for the image processing and pose estimation processes. The *Camera driver* node and its outputs are explained in more detail in Section 6.5.3.

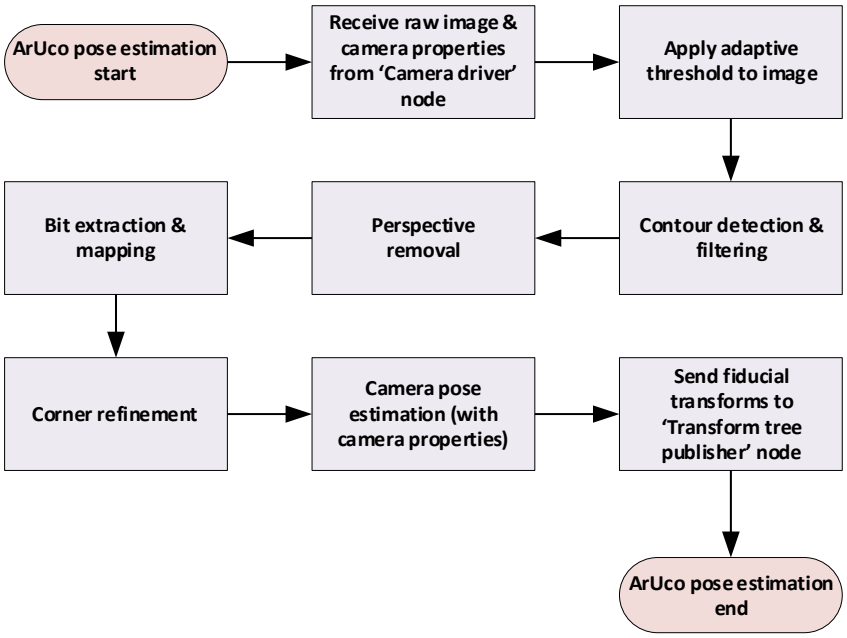


Figure 6.8 - Flow chart describing the process followed by the pose estimation algorithm.

The first step of the image processing applies an adaptive threshold to the image. This step essentially applies maximum contrast to the image and highlights the borders of the objects in the image. The next step involves detecting contours in the image and then filtering the detected contours bases on the perimeter sizes and proportions of the contours. Following this, prospective marker candidates go through a perspective removal process, which essentially flattens out any marker contours which are not parallel to the camera lens. Once the perspective removal has occurred, the now square marker candidates go through a bit extraction process. In this step the potential marker image is divided into a grid with a size equal to the number of bits in the chosen marker dictionary. The grid cells are then mapped according to black and white cells and then compared to markers in the chosen dictionary. If a match is found, the marker is successfully detected and assigned an ID. An optional step allows for refining the corners of detected markers, improving the accuracy of the corner locations of each marker. The corner locations are then ready to be used in the camera pose estimation.

The camera pose estimation step determines the camera pose relative to each detected marker and corrects for distortion introduced by the camera lens. To do this it makes use of the detected marker corner locations, the fiducial side length (entered as a parameter at the start of the experiment), and the camera calibration data found in the camera properties messages. The estimated pose relative to each marker is output as a 3D transformation matrix, which transforms points from its respective marker's coordinate system to the camera coordinate system. An array containing each transformation indexed according to the marker ID (and thus each cart ID) is then sent over the 'Fiducial transforms' topic.

Having the pose of each marker available in the form of the 3D transformation is not practical and not intuitive for understanding by the user, nor for use by the cart payloads. Rather, it is desirable to display, log and send the pose information in the form of a familiar planar coordinate system (x, y and θ) relative to a reference marker. This is achieved through the help of auxiliary software components - the *Transform tree publisher* node and *Cart pose publisher* node.

The *Transform tree publisher* node makes use of the 'tf' ROS library functions (ROS Tf Documentation, 2018). The node manipulates the fiducial transforms to a form which can be used to determine each marker's planar coordinates and rotation relative to a reference marker. The 'tf' ROS library allows for the creation of a tree-like structure of 3D transformations and coordinate systems with included time information. The tree is mapped in a way that allows the transformations to be easily converted between reference coordinate frames. This transform tree is then published to the *Transform tree* topic and can then be used by the *Cart pose publisher* node.

The *Cart pose publisher* node accesses the published transform tree and converts the 3D transformation of each marker (which is referenced to the camera coordinate frame) to a transformation referenced to the origin marker's coordinate system. It then uses some 'tf' ROS library functions to calculate the planar translational and rotational position vectors (x,y and θ) for each cart's marker relative to the origin marker. As mentioned previously, the transform tree also contains time information – for example, it contains the transformations from the previous timestep. This feature is used by the *Cart pose publisher* node to estimate the translational and rotational velocities of each marker using a finite difference approximation. A first-order backward difference approximation is used. After these calculations, the *Cart pose publisher* node organises the planar position and velocity information for each marker into an array indexed according to each cart's unique ID. This array is then published to the *Cart pose info* topic, where the communication manager can relay the data to the cart payloads, display it to the user and log it. The data is published at 10 Hz as required by the demonstration payload's control (described in Chapter 7), but the *ArUco pose estimation* node is able to output pose information at rates up to 60 Hz.

6.5.3 Camera hardware, implementation, and calibration

The camera used is a 1.6 MP colour USB board camera, the DFM 37UX273-ML. The camera is capable of high framerates, up to 238 fps using USB 3.1, with a global shutter. The exposure time can be varied between 1 µs and 30 s. The camera was chosen for its versatility – capable of a large range of framerates and exposure times, as well as producing colour images. Furthermore, the camera has an existing Linux driver and software development kit (SDK) which are helpful for integration with ROS.

Through experimentation a pixel per metre value of more than 250 was seen to be adequate for the detection of 6x6 ArUco markers with side lengths of 6 cm. The camera used has a maximum output resolution of 1440x1080, achieving a theoretical pixel per metre value of 375 at the table's surface, making it more than adequate for detecting similar markers. The camera is run at 30 fps for the case of the demonstration payload but can be run at higher framerates if required by the experiment.

The camera is mounted to the ceiling of the satellite test facility's room, using a custom-designed assembly of 3D printed parts. The assembled mounting apparatus with the camera housed is shown in Figure 6.9. A USB 3.1 cable is used for powering the camera and data transfer. The mounting apparatus allows for manually adjusting the pan and tilt of the camera for adjusting the image scene. An M12 lens and appropriate lens holder are assembled with the board camera and housed in the mounting apparatus. A lens with a focal length of 3.6 mm is used, which allows for the entire glass platform to be in the camera's field of view as seen from the mounting point.

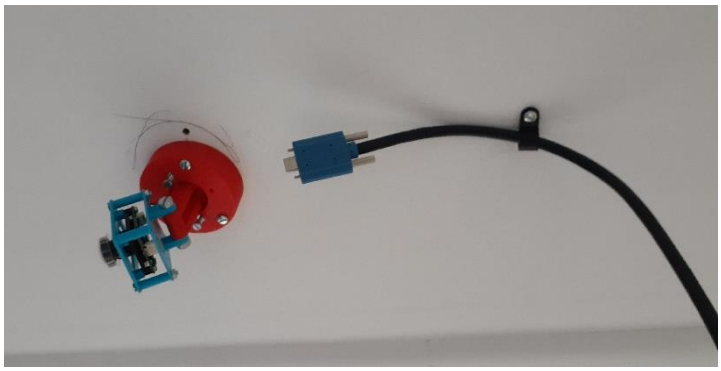


Figure 6.9 - Mounting apparatus used to mount the camera to the ceiling.

An example image taken by the camera in its mounted position is shown Figure 6.10. The camera parameters used during image acquisition, such as gain, exposure time and whitebalance settings were tuned until ArUco detection was

seen to be reliable on most of the glass platform. The values used can be found in Appendix B.1.

Figure 6.10 - An example image showing the view of the glass platform with a carrier cart atop, as seen by the test facility's camera.

The *Camera driver* ROS node is responsible for interfacing the camera with the rest of the optical pose estimator. The camera driver software used is based on GStreamer and is provided in the SDK from the camera manufacturer. The node outputs the raw images at the pre-configured framerate and publishes them to the *Raw images* topic. The node also loads the camera's calibration information from a file (created during the calibration process described in the following paragraphs) and publishes it to the *Camera properties* topic. Both published topics are then used for the ArUco pose estimation algorithm as described previously.

The camera was calibrated using a ROS package based on OpenCV's camera calibration library. The calibration process involved capturing a series of images of a flat checkerboard pattern at various angles, positions, and distances away from the camera. Figure 6.11 shows an example image from the set of calibration images used. The checkerboard pattern used was printed on a hard wooden board to ensure the pattern was as flat as possible. A 4x6 checkerboard pattern was used

for the calibration. The 4x6 dimension refers to the amount of internal corners of the checkerboard pattern.

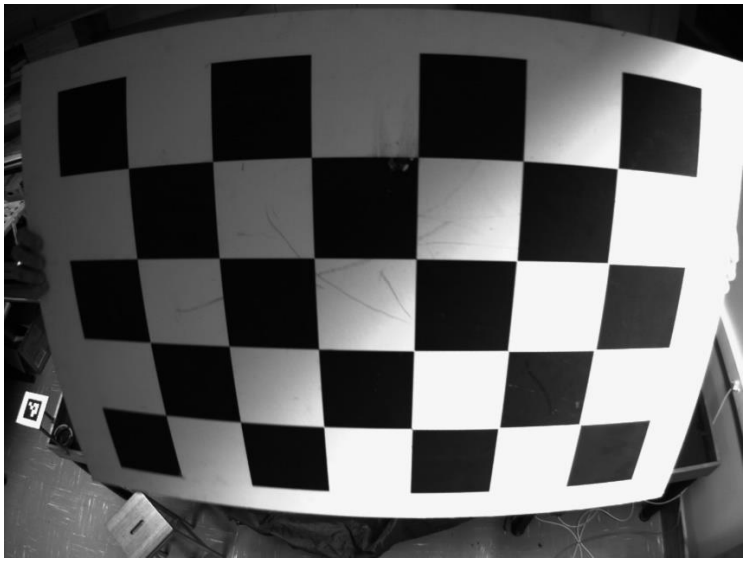


Figure 6.11 - An example image which forms part of the set of calibration images used.

The camera calibration package outputs a file containing information about the camera resolution, the distortion model used, the camera matrix, the distortion coefficients, the rectification matrix, and the projection matrix. All this information is used by the ArUco pose estimation algorithm to mitigate the inaccuracies introduced by the camera lens' distortion. Figure 6.12 shows the same view of the test facility as seen previously in Figure 6.10, but for this image the parameters determined during calibration were used to algorithmically correct the distortion in the image. Comparing the undistorted image to the raw image establishes a visual indication of the results of the camera calibration. One clear indicator is that in the raw image in Figure 6.10, the bottom edge of the table is curved – while in the undistorted image in Figure 6.12, the table edge is straight as it is perceived in reality.

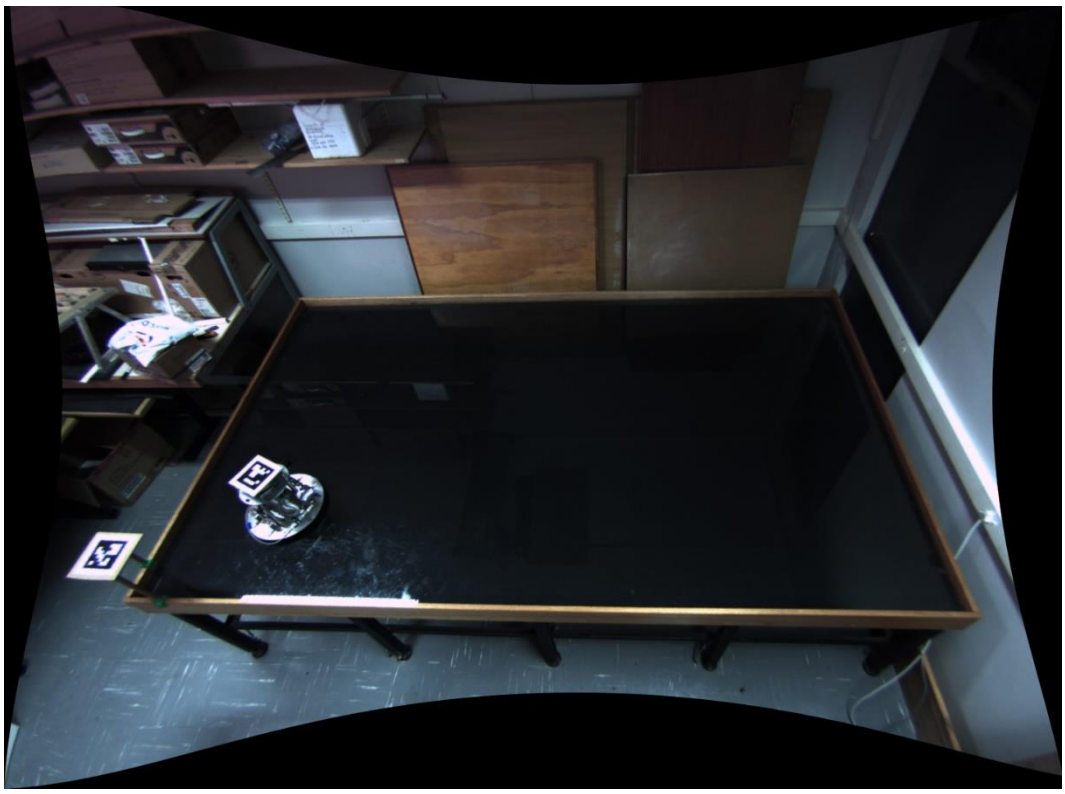


Figure 6.12 - A processed image showing the view from the test facility's camera with correction for lens distortion applied.

6.6 User interface implementation

The user interface allows the user to interact with their experiment by viewing and logging current pose and control data, as well as wirelessly communicating with the carts if desired (for example, live changing of parameters on their payload). Recalling from Figure 6.3, the components of the user interface are reiterated in Figure 6.13. Not shown in Figure 6.13 are the connections made by the ROS tools & utilities, as these ROS elements essentially communicate with all of the ROS nodes including the ROS master node.

The user interface is mostly implemented using a collection of ROS tools and utilities, but for the case of the demonstration payload a custom node - the *Cart command input* node - is used for sending commands to the payloads which are entered by the user during operation. The command messages sent over the *Cart command data* topic has a generalised form that should be able to be directly used for general-purpose experiments if desired. However, if the user wants to send more specific commands to their payload, they will need to edit the source code of this node or write a similar one themselves. This would only require a basic understanding of ROS nodes and ROS messages to do. The rest of the FCS is

designed with the user interface in mind, allowing for the majority of data to be logged and displayed in near-real time, and for many relevant parameters to easily be changed by the user during or before operation.

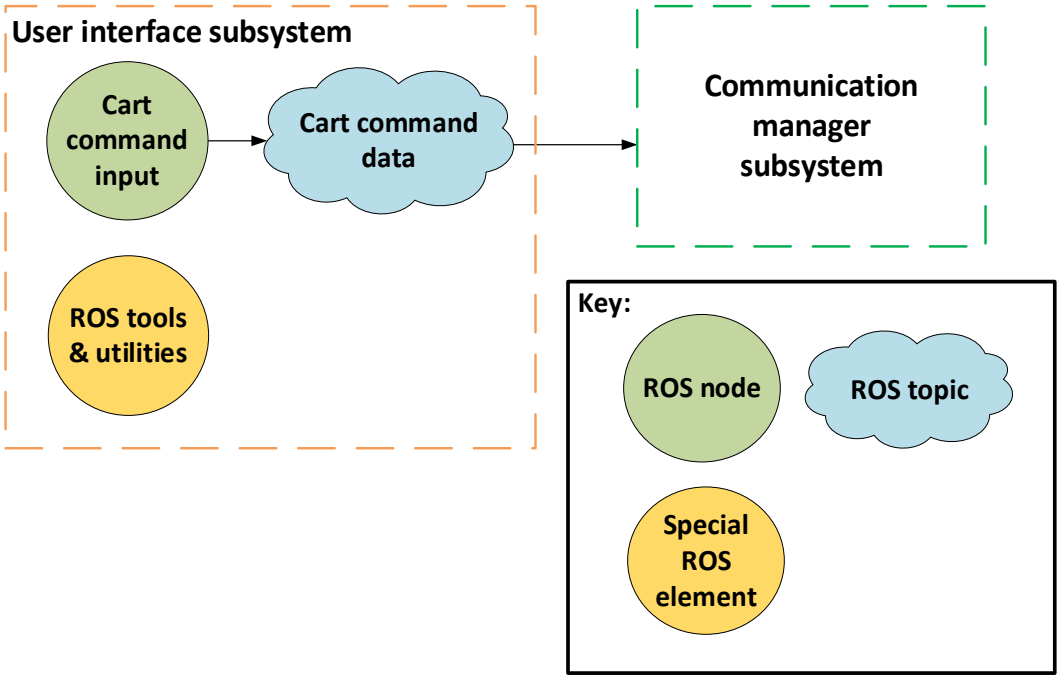


Figure 6.13 - A diagram reiterating the elements of user interface, extracted from Figure 6.3.

The major ROS tools and utilities used for the user interface are the 'rosbag' package, the ROS Qt (RQt) framework, RViz, and the ROS Parameter Server. These are all fundamental parts of the ROS toolkit. The 'rosbag' package is used for data logging, RQt and RViz are used for data visualisation, and the ROS Parameter Server is used for setting and fetching relevant parameters. The role and usage of each will be discussed in the following paragraphs.

The 'rosbag' package allows for logging data that is sent over ROS topics during operation. The package is capable of logging all topics simultaneously, allowing for all communication between nodes during an experiment to be stored in so-called bag files. A bag file is a file format which contains the logs of ROS messages. This data can be played back just as it happened during the experiment or can be viewed in a graphical user interface (GUI) using an RQt plugin. The data in the bag file can also be extracted and used with the MATLAB ROS toolbox, which will be convenient for most users as they will likely have MATLAB experience.

The RQt framework is based on Qt, the open-source GUI toolkit, and facilitates GUI development in ROS. The RQt framework has many useful plugins of which two are mainly used for the user interface. The RQt Plot plugin is used to display near

real-time data being sent over any active ROS topics, and for the case of the user interface it is used to plot the cart pose information versus time. The user can easily choose which elements of which messages should be displayed, and any topic can be accessed – allowing a wide variety of data to be plotted and monitored during the experiment. Figure 6.14 shows a screenshot of the RQt Plot plugin's GUI as an example of how data would be displayed in near-real time to the user.

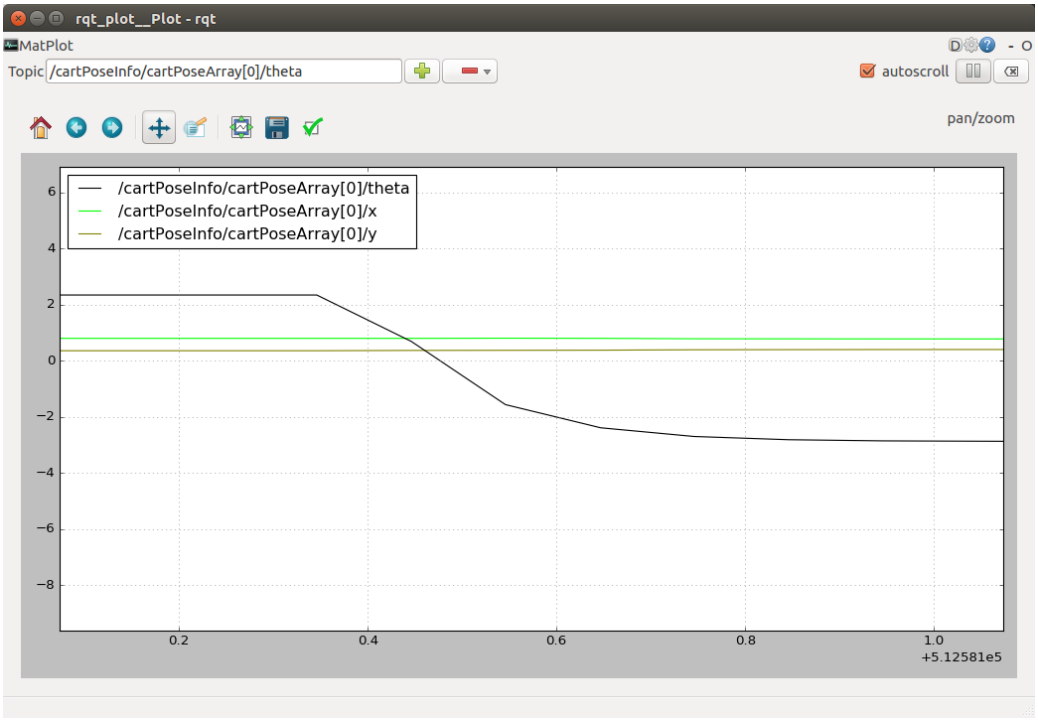


Figure 6.14 - A screenshot of the RQt Plot plugin GUI used as part of the user interface.

Another RQt plugin used is the RQt Bag plugin, which eases the process of interfacing with ROS bag files by means of a GUI. This plugin allows for viewing the data in the bag file in an intuitive way - enabling the user to plot data, play back captured images as a video and view contents of messages at specific timestamps. A screenshot showing the layout of the RQt Bag plugin's GUI is shown in Figure 6.15.

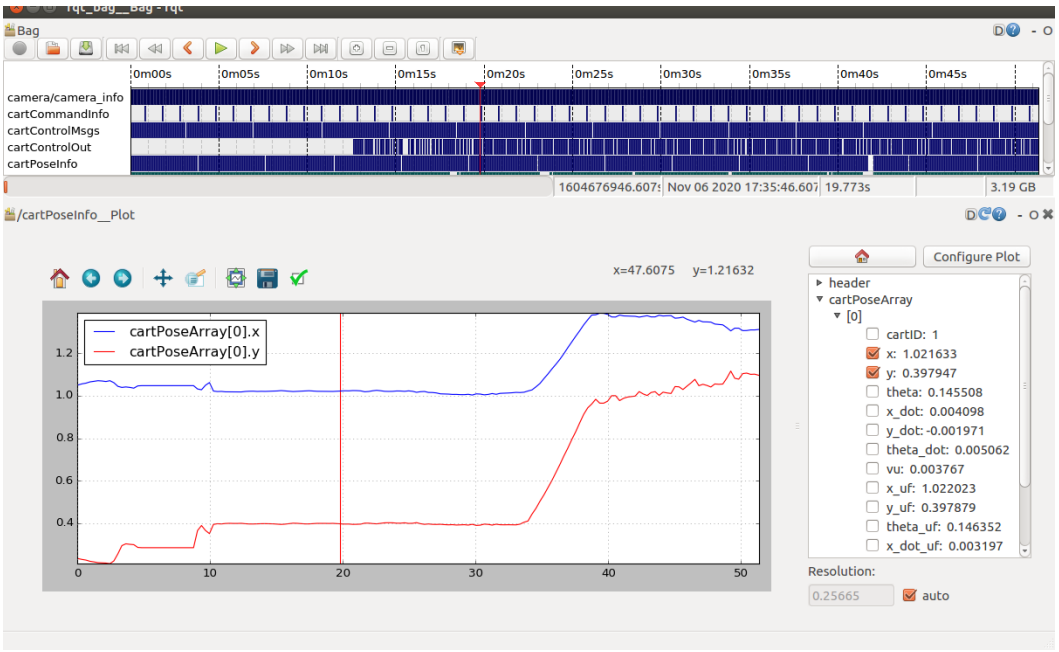


Figure 6.15 - A screenshot of the RQt Bag plugin GUI used to view the contents of log files.

RViz is a 3D visualisation environment built for ROS. Among other capabilities, it can display camera images and the positions and orientations of coordinate frames in a 3D rendered space with gridlines. These are the main two features used for the user interface. Firstly, the test facility camera's image is displayed with detected ArUco markers' estimated pose coordinate systems overlayed. This provides visual confirmation to the user that ArUco markers are being detected successfully. Secondly, the coordinate frames of the camera and each ArUco marker (as estimated by the optical pose estimator) are shown with labels in a 3D rendered space. This allows the user to visually confirm that the positions and orientations of detected markers are correct. Figure 6.16 shows the layout of RViz used for the test facility's user interface as it would look during a typical experiment.

The ROS Parameter Server is managed by the ROS master node and allows for setting and fetching of parameters over the ROS network. The ROS Parameter Server is accessible by all ROS nodes and each node can set and fetch parameters at runtime. The ROS Parameter Server also has built-in command line functionality, allowing the user to easily read and change parameters during operation. The parameters on the server are addressed using ROS's normal hierarchical naming convention, which makes it easy for the user to discern which parameters are associated with which nodes. Using the ROS Parameter Server's command line functions, the user can view the hierarchical list of active

parameters, read their values, and set their values during any stage of the operation.

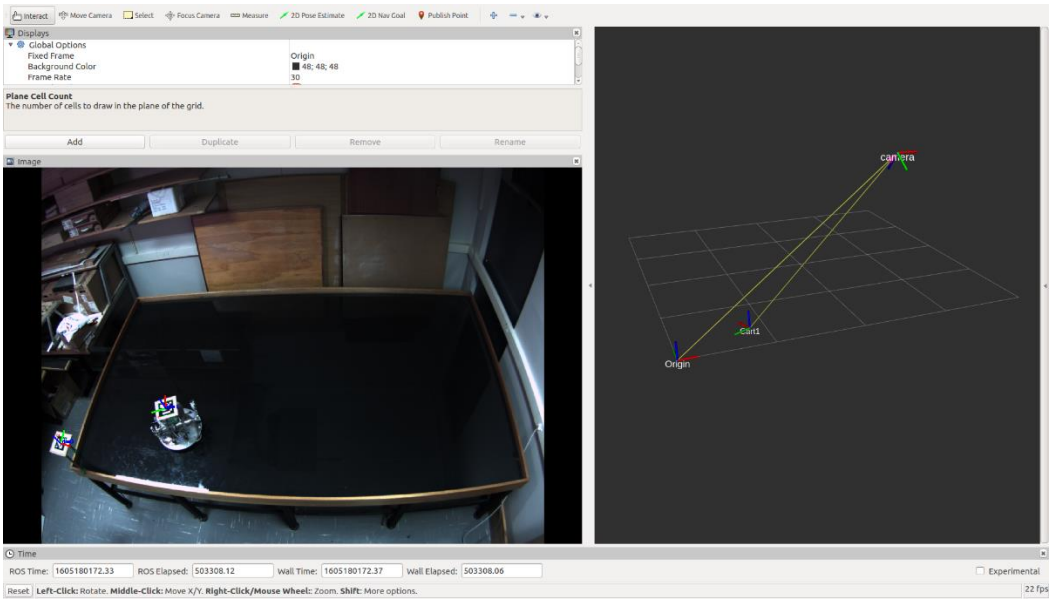


Figure 6.16 - A screenshot of the RViz 3D visualisation tool used as a part of the user interface.

The nature of the FCS's design and the modularity of ROS allows for much potential in terms of easily expanding the user interface for different experiments. For example, the ROS Parameter Server's command line functions are suitable for the case of the demonstration payload where only a few parameters need to be set during operation – but this may be tedious for more complex experiments where more parameters need to be set more often. The RQt framework can once again be used to set the parameters using a GUI, which will be more convenient for such a case. In fact, an RQt plugin for this exists and can easily be integrated with the user interface by simply running it. The use of open-source and modular software toolkits for the FCS makes this simple integration possible.

6.7 Control data output of the demonstration payload

Since the demonstration payload is meant to showcase the test facility's capabilities, it was designed to make use of the FCS's bi-directional wireless communication functionality. It does this by receiving pose information from the system for control feedback, while simultaneously communicating its controller outputs and other relevant data to the system to allow for it to be displayed and logged. To achieve this, the ROS core libraries were ported and implemented on the ESP8266 microcontroller with the help of the 'rosserial' ROS package (ROS Rosserial Documentation, 2018).

Aside from sending data to the carrier cart payload as discussed in Section 6.4, the FCS is also capable of receiving ROS messages published by the payload. The payload acts as a pseudo ROS node – the only difference being that the payload does not communicate directly with the ROS master node, but rather communicates with the *Wireless server* node which forms part of the communication manager subsystem. The implementation of ROS on the demonstration payload is described in Figure 6.17, which expands upon a portion of the communication manager subsystem shown in Figure 6.3 and shows the details relevant to the demonstration payload. The *Cart control output* topic was omitted from Figure 6.3 previously because not all experiments will feature this node – it is dependent on whether the user makes use of the bi-directionality of the test facility's wireless communication.

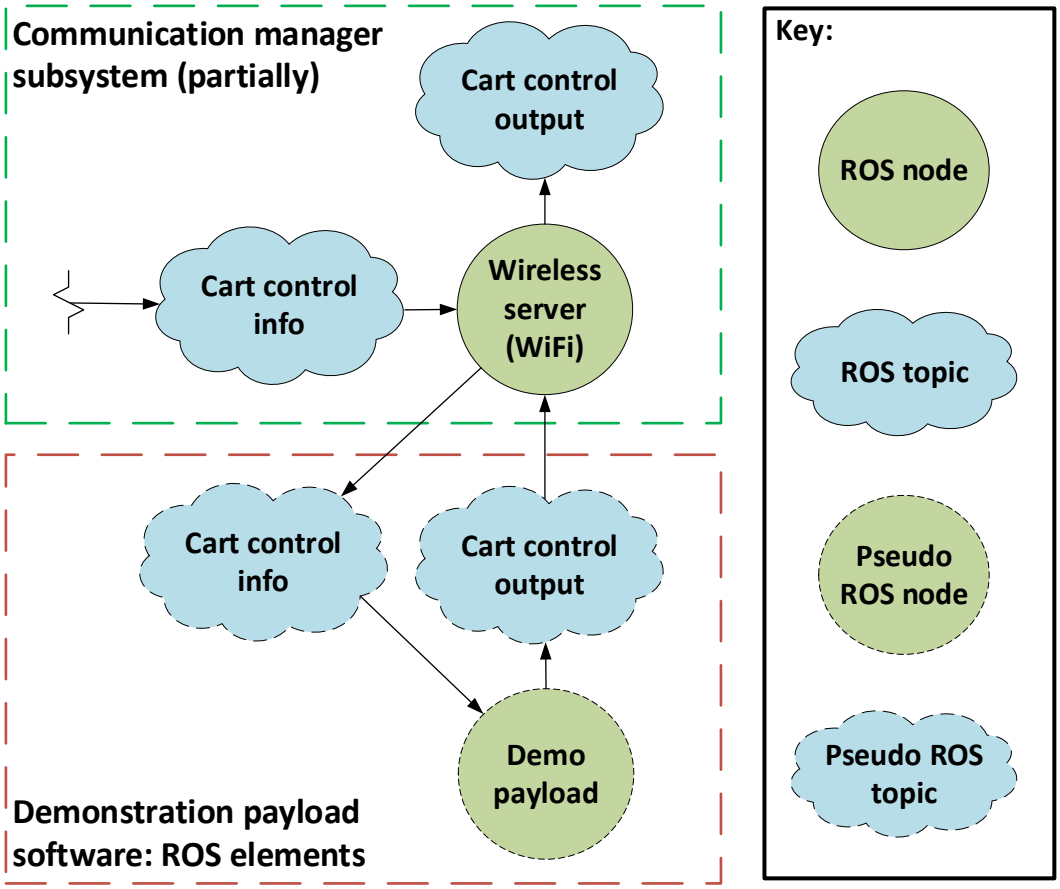


Figure 6.17 - A diagram describing the ROS implementation on the demonstration payload.

The pseudo ROS node and pseudo ROS topics in Figure 6.17 are named as such because the ROS implementation on the demonstration payload's ESP8266 microcontroller is not technically part of the ROS network. The data sent and received by the payload is transferred as raw data over WiFi, but the

demonstration payload's software organises this internally in a similar way to ROS messages and topics. This data is then relayed to or from actual ROS topics created by the *Wireless server* node. This all facilitates communication between the payload and the ROS master and thus the communication manager subsystem.

The demonstration payload publishes its control data over the *Cart control output* topic. This data includes control signal outputs, runtime metrics, calculated control parameters, and received pose information. This data can then be displayed and logged through the user interface.

6.8 Summary and conclusion

This chapter discussed the design and implementation of the test facility's feedback and communication subsystem. The FCS is designed to optically estimate the pose of the carrier carts on the glass platform, while also logging, displaying, and wirelessly communicating this data to the carrier cart payloads. The software elements of the FCS are implemented using the open-source robotics framework known as ROS. The use of an open-source software framework adds to the versatility and expandability of the test facility and will likely improve its usability for users.

With the carrier cart and the FCS designed and implemented, the test facility is ready to operate. To demonstrate the test facility's functionality, a closed-loop control system was designed to be run on the demonstration payload. The following chapter discusses the design of the control system, as well as its implementation on the demonstration payload.

7 Control system design

This chapter focuses on the control system of the demonstration payload, which was designed to demonstrate the functioning of the satellite test facility. It looks at the mathematical model of the carrier cart's motion on the table, as well as the design, simulation, and implementation of the payload's control system.

7.1 Dynamic model of the carrier cart

Figure 7.1 shows a free body diagram of the carrier cart. The coordinate systems used for the model of the cart are divided into two sets – the fixed coordinate frame of the glass table and the body coordinate system of the carrier cart. The $\bar{x}$ and $\bar{y}$ axes are fixed relative to the facility's glass table; while the $\bar{x}_B$ axis is the heading direction of the cart, which changes as the cart rotates. The angle between the $\bar{x}$ axis and the $\bar{x}_B$ axis is defined as θ, the heading angle. Thrusters are labelled N1 to N4, with the centre of mass also marked as CoM. The thrusters are all assumed to be mounted a distance *L/2* from the centre of mass, and the air bearing is assumed to be coincident with the centre of mass as designed.

The forces F_1 through F_4 correspond to the actuator forces from each thruster. A torque is also imposed by each thruster force through the moment arm *L/2*. The friction force F_f acts on the air bearing, opposite to the direction of the cart's translational velocity. The friction torque T_f also acts on the air bearing, opposite to the direction of the cart's rotational velocity.

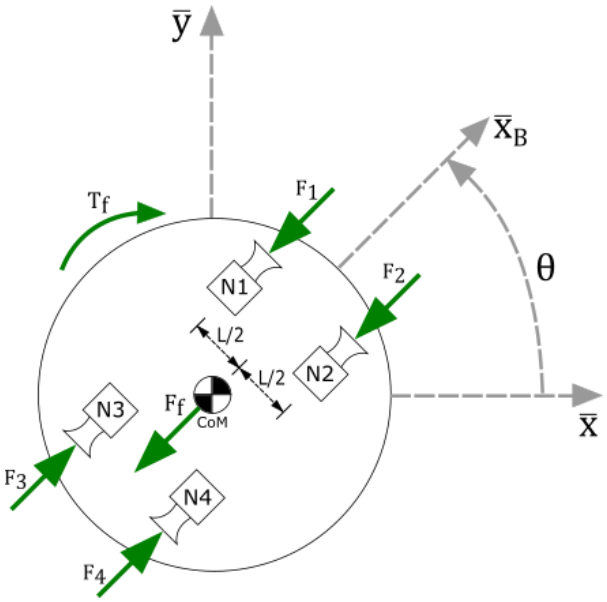


Figure 7.1 - Free body diagram of the carrier cart, showing all forces and torques considered while modelling the dynamics of the carrier cart.

In the following subsections, the plant dynamics for each direction will be derived. The translational dynamics in the heading direction, along with the rotational dynamics will be used for the design of the low-level controllers. The non-linear $\bar{x}$ and $\bar{y}$ dynamics, and the rotational dynamics, will be used for simulating the behaviour of the full cart controller.

7.1.1 Translational dynamics of the carrier cart in the heading direction

Applying Newton's second law in the $\bar{x}_B$ direction,

$$F_3(t) + F_4(t) - F_1(t) - F_2(t) - F_f = m_c \ddot{x}_B(t) \tag{7.1}$$

is obtained, where m_c is the mass of the cart, and $\ddot{x}_B$is the acceleration of the cart in the heading direction $\bar{x}_B$. The friction force F_f is defined as

$$F_f = b_{\mathrm{AB}} \dot{x}_B(t), \tag{7.2}$$

where b_{AB} is the friction coefficient of the air bearing and $\dot{x}_B$ is the velocity of the cart in the heading direction $\bar{x}_B$. Since the actuators will be controlled as an input to the plant, it is convenient to group them as single a time-varying controllable force in the $\bar{x}_B$ direction. This force F_T is defined as

$$F_T(t) = F_3(t) + F_4(t) - F_1(t) - F_2(t). \tag{7.3}$$

Substituting Equations (7.2) and (7.3) into Equation (7.1) and rearranging yields

$$m_c \ddot{x}_B(t) + b_{\mathrm{AB}} \dot{x}_B(t) = F_T(t). \tag{7.4}$$

Applying the Laplace transform, assuming zero initial conditions, to both sides of Equation (7.4) and rearranging yields the continuous domain plant transfer function in the $\bar{x}_B$ direction:

$$H_{x_B}(s) = \frac{X_B(s)}{F_T(s)} = \frac{1}{m_c s^2 + b_{\mathrm{AB}} s}. \tag{7.5}$$

7.1.2 Rotational dynamics of the carrier cart

Applying Newton's second law in rotational form for the cart,

$$\big(F_1(t) + F_4(t) - F_2(t) - F_3(t)\big)\frac{L}{2} - T_f = J_c \ddot{\theta}(t) \tag{7.6}$$

is obtained, where J_c is the mass moment of inertia of the cart, and $\ddot{\theta}$ is the angular acceleration of the cart in the positive θ direction. The friction torque T_f is defined as

$$T_f = B_{\mathrm{AB}}\dot{\theta}(t), \tag{7.7}$$

where B_{AB} is the rotational friction coefficient of the air bearing and $\dot{\theta}$ is the angular velocity of the cart in the positive θ direction. Similarly to the grouping of the actuator output forces in Section 7.1.1, it is convenient to group the torques induced by the thruster forces as a single time-varying controllable torque in the θ direction. This torque T_T is defined as

$$T_T(t) = (F_1(t) + F_4(t) - F_2(t) - F_3(t))\frac{L}{2}. \tag{7.8}$$

Substituting Equations (7.7) and (7.8) into Equation (7.6) and rearranging yields

$$J_c\ddot{\theta}(t) + B_{\mathrm{AB}}\dot{\theta}(t) = T_T(t). \tag{7.9}$$

Applying the Laplace transform, assuming zero initial conditions, to both sides of Equation (7.9) and rearranging yields the rotational plant transfer function in the continuous domain:

$$H_\theta(s) = \frac{\theta(s)}{T_T(s)} = \frac{1}{J_c s^2 + B_{\mathrm{AB}}s}. \tag{7.10}$$

7.1.3 Dynamics of the carrier cart in the $\bar{x}$–$\bar{y}$ plane

Applying Newton's second law in the $\bar{x}$ direction yields the equation and

$$(F_3(t) + F_4(t) - F_1(t) - F_2(t))\cos\theta - F_{fx} = m_c\ddot{x}(t). \tag{7.11}$$

Here $\ddot{x}$ is the acceleration of the cart in the $\bar{x}$ direction, and F_{fx} is the friction force in the $\bar{x}$ direction. The friction force is modelled as

$$F_{fx} = b_{\mathrm{AB}}\dot{x}(t). \tag{7.12}$$

Grouping the actuator forces by substituting Equation (7.3) and Equation (7.12) into Equation (7.11) and rearranging yields

$$m_c\ddot{x}(t) = F_T\cos\theta - b_{\mathrm{AB}}\dot{x}(t), \tag{7.13}$$

which describes the dynamics of the cart in the $\bar{x}$ direction.

Following a similar process, the dynamics of the cart in the $\bar{y}$ direction are described by

$$m_c\ddot{y}(t) = F_T\sin\theta - b_{\mathrm{AB}}\dot{y}(t). \tag{7.14}$$

Rearranging Equation (7.9) yields a similar differential equation describing the rotational dynamics of the cart as

$$J_c\ddot{\theta}(t) = T_T(t) - B_{\mathrm{AB}}\dot{\theta}(t). \qquad (7.9^*)$$

The differential Equations (7.9), (7.13) and (7.14) are used to model the dynamics of the cart on the table given the output thrust signals from the controller. Since the dynamics in the $\bar{x}$ and $\bar{y}$ direction are described by non-linear differential equations, it is more convenient to model them based on the differential equations rather than an s-domain transfer function.

7.2 Simulink model of the carrier cart dynamics

Figure 7.2 shows a block diagram of the plant which represents the carrier cart in the Simulink model. The inputs to the plant are the thruster control signals which will be determined by the controller, while the output of the plant is the state of the cart in terms of x, y and θ including positions and velocities.

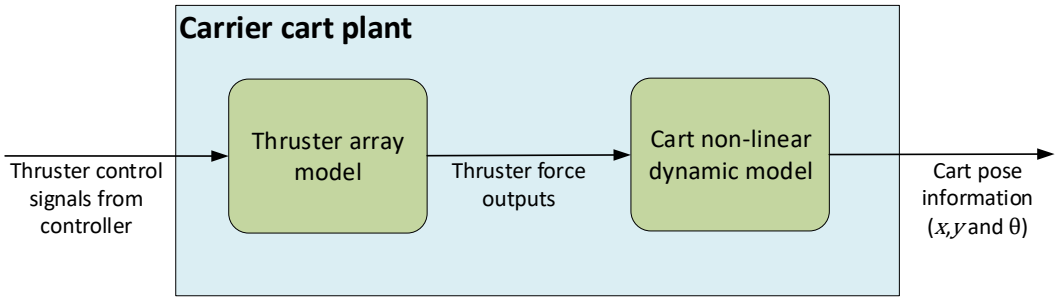


Figure 7.2 - Block diagram of the carrier cart plant as modelled in Simulink.

The thruster array model shown in Figure 7.2 accounts for the response of each of the solenoid valves to their respective control signals. This model includes the delayed opening and closing of the valves, as well as the minimum on and off time of the valve. The time values used for the opening, closing and minimum on and off time were all determined to be 9 ms, as measured during the thruster characterisation experiments in Chapter 5.

Figure 7.3 shows the Simulink implementation for a single thruster model inside the thruster array model. The control signal input, $\gamma_N(t)$, is a digital logic input, which is routed through an on delay and an off delay. These blocks simulate the delayed opening and closing of the solenoid valve, as well as the minimum on and off time of the valve. A gain is then applied to the signal with the value of the thrust produced by the nozzle (F_{Nozzle}) thus outputting the thrust value, $F_N(t)$, to the dynamic model of the plant. $F_N(t)$ represents the general case of one of the four

actuator forces, $F_1(t)$ through $F_4(t)$. The thruster array model has four of these single thruster models corresponding to the four thrusters on the physical cart.

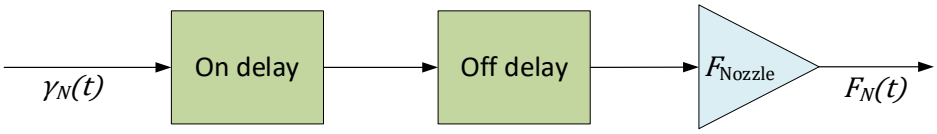


Figure 7.3 - Simulink implementation of a single thruster model.

The non-linear dynamic model shown in Figure 7.2 is based on the dynamics derived in Section 7.1.3. Figure 7.4 shows an overview of this portion of the model. Firstly, the control force, $F_T(t)$, and control torque, $T_T(t)$, are calculated using the four actuator forces $F_1(t)$ through $F_4(t)$. Secondly, the cart state is calculated using the differential equations derived in Section 7.1.3. Figure 7.5, Figure 7.6 and Figure 7.7 show the details of the state calculations for $\theta(t)$, $x(t)$, and $y(t)$ respectively. It is noteworthy that the calculations for $x(t)$ and $y(t)$ require the value of $\theta(t)$.

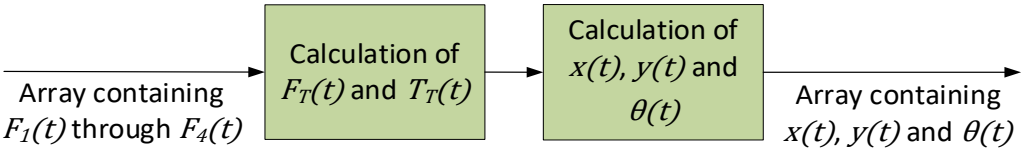


Figure 7.4 - Overview of the non-linear dynamics portion of the Simulink model.

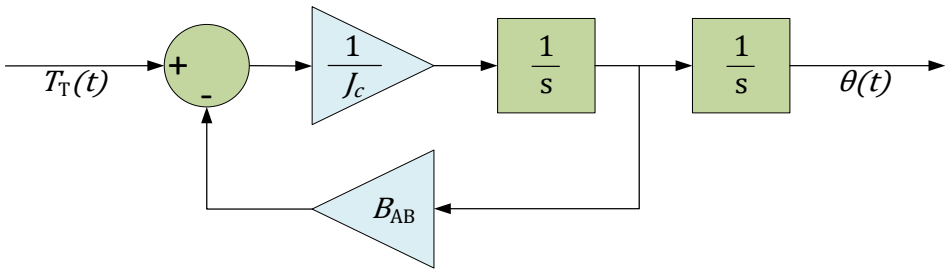


Figure 7.5 - Simulink implementation of the state calculation of $\theta(t)$, which is based on Equation (7.9).

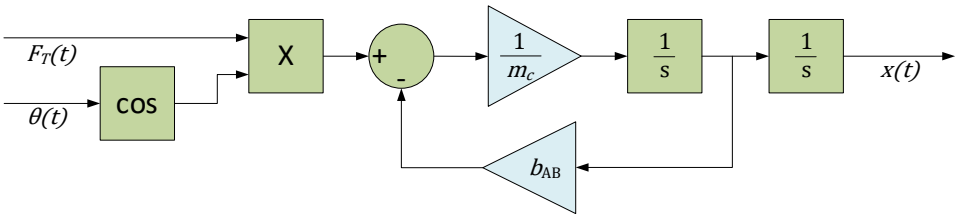


Figure 7.6 - Simulink implementation of the state calculation of $x(t)$, which is based on Equation (7.13).

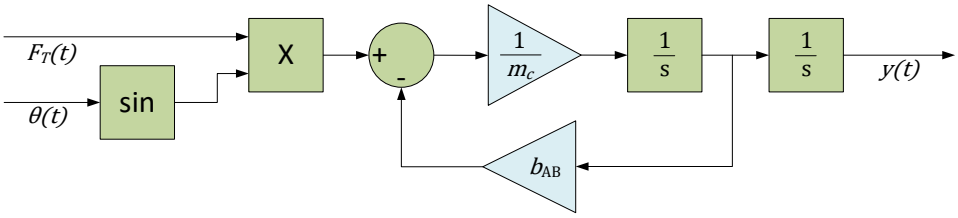


Figure 7.7 - Simulink implementation of the state calculation of $y(t)$, which is based on Equation (7.14).

7.3 Control architecture

The nature of the problem of controlling the carrier carts (with the 'H' thruster configuration, as described in Subsection 4.4.3) is similar to that of the commonly encountered control of a differential drive robot. Consequently, an architecture was chosen which controls heading direction and the speed of the cart in that direction. A diagram describing the control architecture is shown in Figure 7.8.

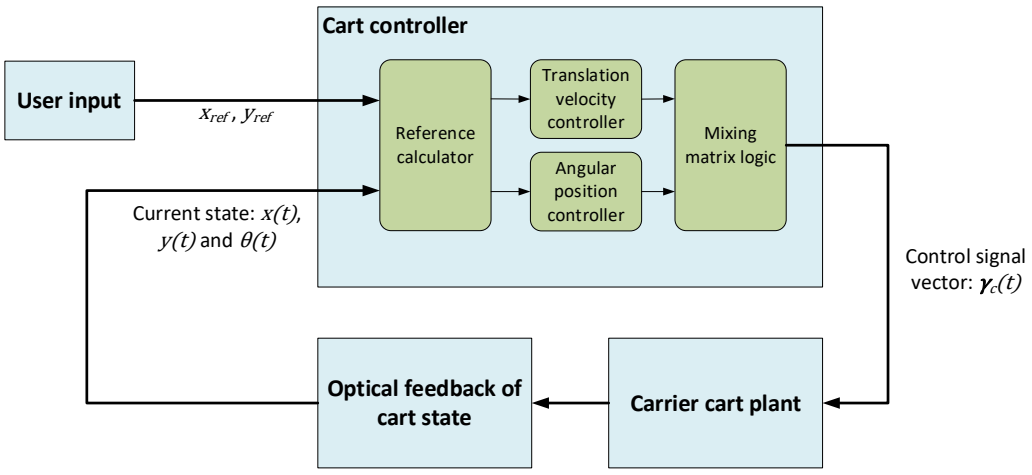


Figure 7.8 - Block diagram describing the control architecture used to control the carrier cart.

The inputs to the controller include the current state of the cart on the table, as well as a user-defined point on the table with the coordinates (x_{ref}, y_{ref}) which the cart should move to. A vector of four control signals, $\boldsymbol{\gamma_c}(t)$, is output by the controller. Each of the four control signals corresponds to one of the four thrusters on the cart. The thrusters move the physical cart, while its current state is determined by the optical positioning system and fed back to the controller.

The subcomponents within the cart controller include two low-level controllers, one which controls the heading direction (angular position), while the other controls the translational velocity in the heading direction. Another component is the reference calculator, which uses the user-input reference coordinates and the current cart state to calculate reference values for each of the two low-level controllers. Finally, the controller makes use of a mixing matrix to determine the control signal vector required given the outputs from the low-level controllers. A more detailed look at each subcomponent follows.

7.3.1 Reference calculator

The reference calculator uses x_{ref} and y_{ref}, along with $\theta(t)$, $x(t)$ and $y(t)$ to calculate reference values θ_{ref} and v_{ref} for the angular position controller and translation velocity controller respectively. The cart is controlled in a sequenced control mode. This means that the controller first rotates the cart to a certain heading, allows the cart to settle, and then translates the cart in the heading direction to reach the target coordinates. A flow diagram describing the role and functioning of the reference calculator is shown in Figure 7.9.

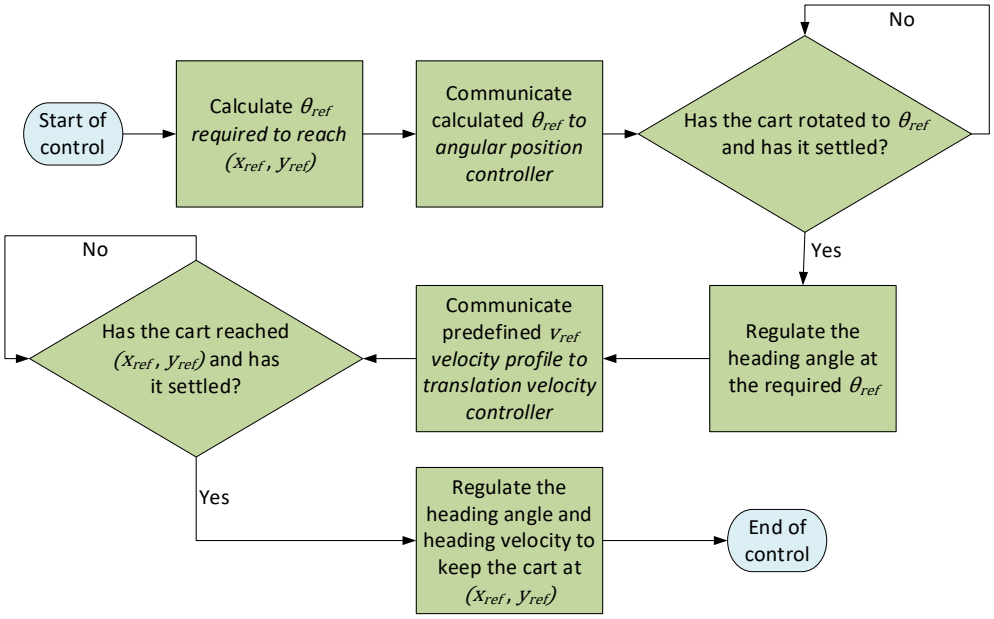


Figure 7.9 - Flow diagram describing the functioning of the reference calculator.

The reference value θ_{ref} is calculated as

$$\theta_{ref} = \tan^{-1}\left(\frac{y_{ref} - y(t)}{x_{ref} - x(t)}\right), \tag{7.15}$$

where x_{ref} and y_{ref} are the target coordinates provided by the user. $x(t)$ and $y(t)$ are the current x and y positions of the cart.

The predefined v_{ref} profile referred to in Figure 7.9 is a velocity profile chosen for the cart to follow as it moves to the target coordinates. The velocity profile chosen is plotted in Figure 7.10. On the plot, v_{max} is a predefined maximum velocity at which the cart will travel, t_θ is the time at which the angular position controller settles at θ_{ref}, and t_p is the time at which the cart reaches the target coordinates within a predefined proximity. The value of v_{max} is directly chosen as a parameter, while t_θ and t_p are indirectly chosen based on other defined parameters.

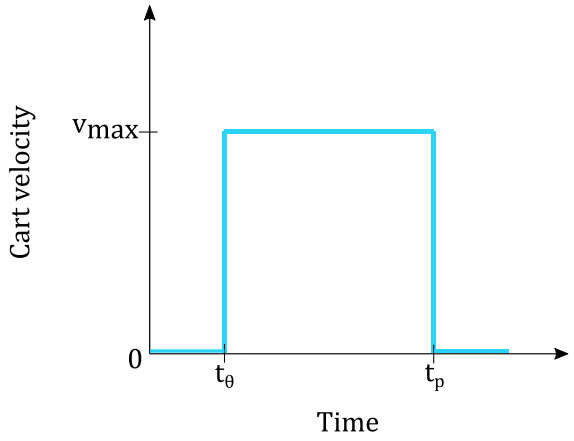


Figure 7.10 - Velocity profile chosen for the cart to follow as it translates to the target coordinates.

The point in time t_θ is defined as the time at which the cart's angular position controller settles at θ_{ref}, which is based on two criteria:

- Is the absolute value of the current angular position error $\theta_{error}(t)$ smaller than a chosen parameter, θ_{thresh}?
- Is the absolute value of the rate of change of $\theta(t)$ smaller than a chosen parameter, Δ_{thresh}?

The instant in time where the above criteria are first met is t_θ, which will be different for each motion of the cart. The angular position error is calculated as

$$\theta_{error}(t) = \theta_{ref} - \theta(t). \tag{7.16}$$

The other time instant, t_p, is defined as the time at which the cart reaches the target coordinates within a circular proximity with a radius of a chosen parameter d_{thresh}. The distance between the cart and the target coordinates is

$$d_c(t) = \sqrt{(x_{ref} - x(t))^2 + (y_{ref} - y(t))^2}\,. \tag{7.17}$$

The time instant where $d_c(t)$ first falls below d_{thresh} is defined as t_p. At this point in time, the cart is considered be in close enough proximity to the target coordinate, and thus according to the velocity profile it is controlled to slow down to a stop.

The parameters v_{max}, θ_{thresh}, Δ_{thresh} and d_{thresh} are design parameters which will affect the behaviour and performance of the cart controller. The final values chosen for the simulated cart controller are listed in Table 7.1. However, just as with other control parameters, the values will need to be tuned for the practical implementation.

Table 7.1 – Chosen values of the reference calculator parameters for use with the simulated controller.

Control parameter	Final chosen value for simulations
v_{max}	0.05 m/s
θ_{thresh}	0.01 rad
Δ_{thresh}	0.02 rad/s
d_{thresh}	0.04 m

7.3.2 Mixing matrix logic

The mixing matrix is responsible for converting the analogue outputs of the two low-level controllers into a vector of four control signals which will achieve those desired outputs. Since the thrusters are discrete and can only be turned on or off, the controller will use pulse-width modulation (PWM) signals to imitate an analogue output. The function of the mixing matrix includes determining which thrusters to activate to apply a force or torque to the cart in a certain direction, as well as calculating the PWM duty cycle required for each thruster.

The logic used to decide which thrusters to activate is shown in Table 7.2. In some cases, the controller may need to apply a torque and a force at the same time. When this happens, three thrusters will be activated to achieve the desired result.

A downside of the chosen thruster configuration is that when this happens, both the torque and force that is used for the control effectively halves.

Table 7.2 - Logic used by the mixing matrix to determine which thrusters to activate to achieve a given force or torque to the cart.

Action required	Thrusters activated			
	N1	N2	N3	N4
Apply no force or torque				
Translate in the positive $\bar{x}_B$ direction			X	X
Translate in the negative $\bar{x}_B$ direction	X	X		
Rotate in the positive θ direction	X			X
Rotate in the negative θ direction		X	X	

A PWM frequency of 10 Hz is used by the controller, as it is compatible with both the solenoid valve's minimum on-time as well as the control rate. The PWM duty cycle to output the required force is calculated as

$$Z_F = \frac{F_C(t)}{2F_{Nozzle}}, \tag{7.18}$$

where $F_C(t)$ is the force output commanded by the translation velocity controller. Similarly, the PWM duty cycle to output the required torque is calculated as

$$Z_T = \frac{T_C(t)}{F_{Nozzle}L}, \tag{7.19}$$

where $T_C(t)$ is the torque output commanded by the angular position controller.

Due to the minimum on-time of the solenoid valve, any duty cycle less than 0.1 will not open the valve. Because of this, if a required duty cycle of less than 0.1 is calculated, the duty cycle is dropped to zero to avoid unnecessarily stressing the solenoid valve. If a required duty cycle is calculated to be larger than 1, it is changed to 1 as the actuator is saturated at this point.

7.4 Low-level controller design

7.4.1 Design methodology

The heading velocity and angular position controllers will be designed in the continuous domain, using the classical frequency-response design method as outlined by Franklin et. al (2015). Since the controllers will be implemented on a microcontroller, they will then be converted to discrete equivalents.

Each controller will be designed and simulated separately as single-input single-output (SISO) systems in this section, while in Section 7.5 both will be combined and the cart controller will be simulated as a whole.

7.4.2 Angular position controller

A lead compensator controller was chosen for the angular position controller, to achieve a relatively fast and well-damped control for the rotation of the cart. A block diagram describing the controller architecture used for the design and simulation is shown in Figure 7.11. The rotation plant is based on the transfer function described by Equation (7.10). An actuator limiter block accounts for the minimum on-time of the solenoid valve, as well as the maximum force output of the thruster. The zero-order hold simulates the sampling of the cart's state at the sampling frequency of the optical feedback system.

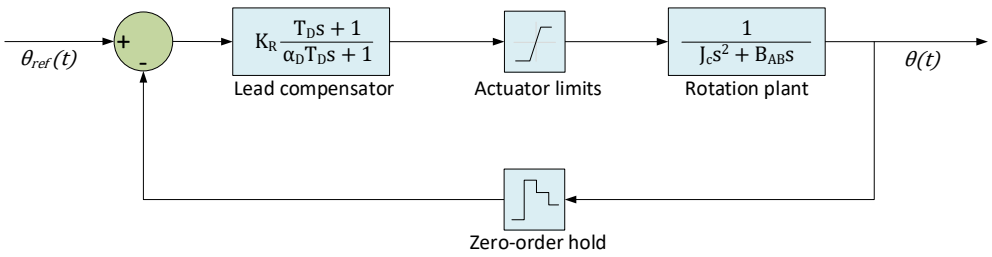


Figure 7.11 - Block diagram showing the control architecture used for the design and simulation of the angular position controller.

The transfer function of the controller is of the form

$$D_R(s) = K_R \frac{T_D s + 1}{\alpha_D T_D s + 1}, \tag{7.20}$$

where K_R, α_D and T_D are design parameters. For a lead compensator, $\alpha_D < 1$. The parameter α_D affects the maximum phase contribution while the T_D parameter

affects the frequency at which the maximum phase contribution occurs for a given α_D value.

Firstly, the value of the gain K_R was chosen to achieve a desired rise time of the closed loop step response. A rise time of less than 10 s was aimed for. Next, the value of α_D was chosen to achieve a phase contribution which yields an overshoot percentage less than 5%. Finally, T_D was assigned a value to place the maximum phase contribution at the natural frequency to achieve optimal damping. The final values chosen for the design are shown in Appendix C.1.

Figure 7.12 shows the simulated non-linear response of the angular position controller to a step input. The simulated controller meets the design specifications, achieving an overshoot of less than 1% and a rise time of 7.4 s. Some minor oscillations due to limit cycling is observed after the controller has settled, which is caused by actuator limitations. This does not affect the control significantly and will be tuned out if it remains a problem in the practical implementation.

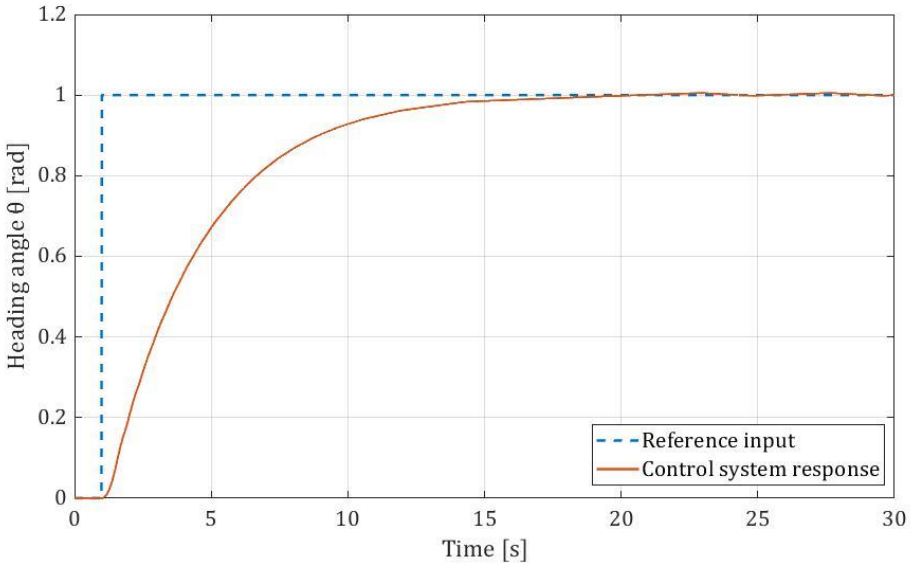


Figure 7.12 - Step response of the angular position controller.

7.4.3 Heading velocity controller

A lag compensator was chosen to achieve a good steady-state response for the heading velocity controller. Proportional control alone yielded an acceptable response for the case where the actuator limits were not considered, but the minimum on-time limitation causes the steady state error to be significant. This warranted using a lag compensator, even though it slightly increases overshoot.

Integral control was considered, but integer windup would be a significant problem because of the actuator limitations – so a lag controller is used instead. Figure 7.13 shows a block diagram describing the control architecture used for designing and simulating the heading velocity controller. The translation plant transfer function is based on Equation (7.5). The transfer function of the lag compensator controller is of the form

$$D_R(s) = K_T \alpha_I \frac{T_I s + 1}{\alpha_I T_I s + 1}, \tag{7.21}$$

where K_T, α_I and T_I are the controller design parameters. For a lag compensator, $\alpha_I > 1$. The parameter α_I affects the low frequency gain and the amount of phase decrease, while the T_I parameter influences the frequency at which the phase decrease occurs. The controller is designed to increase the low frequency gain enough to achieve an acceptable steady state error, while ensuring that the phase decrease introduced does not significantly lower the phase margin (and thereby increasing overshoot). The final values used for the design are listed in Appendix C.1.

Firstly, the gain value K_T was chosen to achieve an acceptable rise time and overshoot. A rise time of less than 1 second, with an overshoot of less than 10% is desired. Secondly, the value of α_I was chosen to achieve a desired steady state error of less than 10% by increasing the low frequency gain. Finally, T_I was assigned a value which places the phase reduction frequency one decade below the system's crossover frequency to minimise the added overshoot.

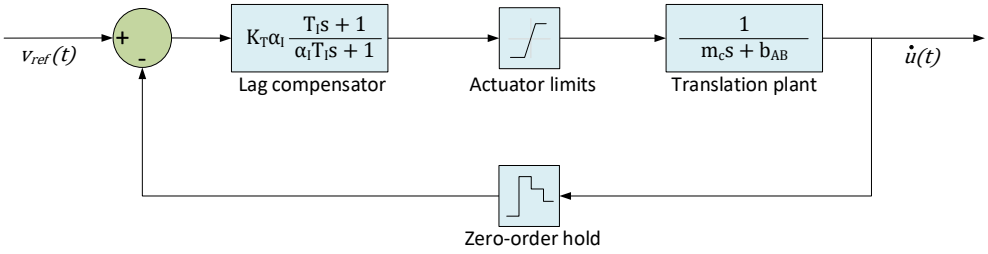


Figure 7.13 - Block diagram showing the control architecture used for the design and simulation of the heading velocity controller.

The simulated step-up and step-down response of the designed heading velocity controller is plotted in Figure 7.14. This form of reference input was chosen because it matches a typical expected case of the reference velocity profile discussed in Subsection 7.3.1. The desired specifications are met in the simulation - as the designed controller achieves a rise time of 0.4 s, with an overshoot of 7.8% and a steady state error of less than 1.4%. The response of the heading velocity

controller is seen to have a stepped-shape because of the minimum on-time of the thrusters, which puts a limit on the smallest adjustments to the velocity.

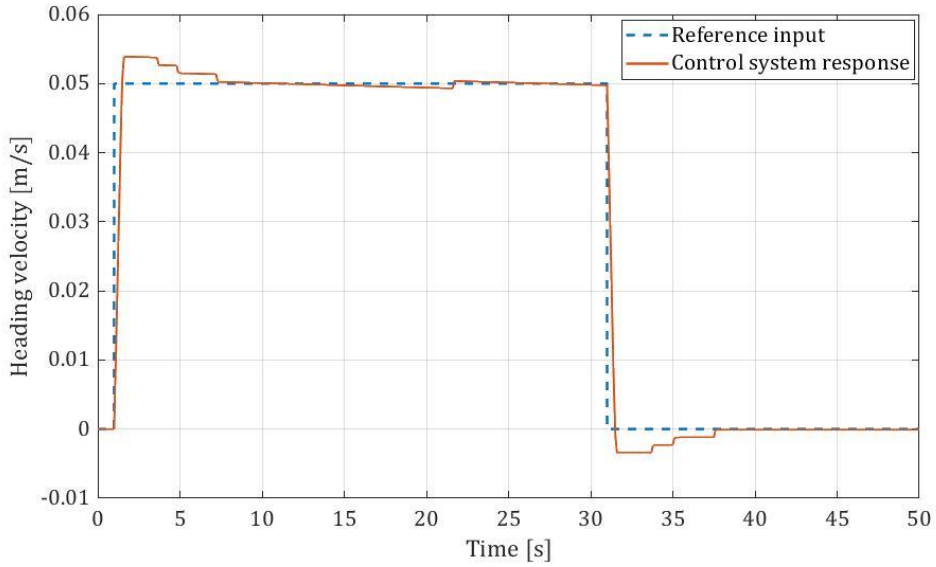


Figure 7.14 - Step-up and step-down response of the heading velocity controller.

Both controllers met their desired specifications when simulated as SISO systems. The following section will show and discuss simulated results of the full cart controller, where both low-level controllers work in tandem along with the cart controller's other subsystems to control the cart's motion.

7.5 Control simulation results

7.5.1 Controller specifications and simulation overview

The desired specifications for the cart controller are simply based on overshoot and steady state error. The design aims for an overshoot of less than 10%, as well as a steady state error of less than 0.05 m in the $\bar{x}$ and $\bar{y}$ direction. Since the cart is meant to emulate a satellite in orbit, where control manoeuvres are typically slow, rise time and settling time are not of much concern. Another requirement of the design is that less than 17 s of thrust time should be used for a given step input manoeuvre with a position change of 1 m and a rotation change of 180°. This will allow at least 4 manoeuvres to be done before running out of propellant when starting with a full tank.

Figure 7.15 shows a block diagram describing the overview of the Simulink model used to simulate the cart controller's functioning. As indicated in the figure, the

cart controller subsystem's inner workings are explained in more detail in Sections 7.3 and 7.4. Likewise, the details on the Simulink implementation of the carrier cart plant are discussed in Section 7.2. The optical feedback system is modelled as a zero-order hold to emulate the digital sampling of the optical pose estimator.

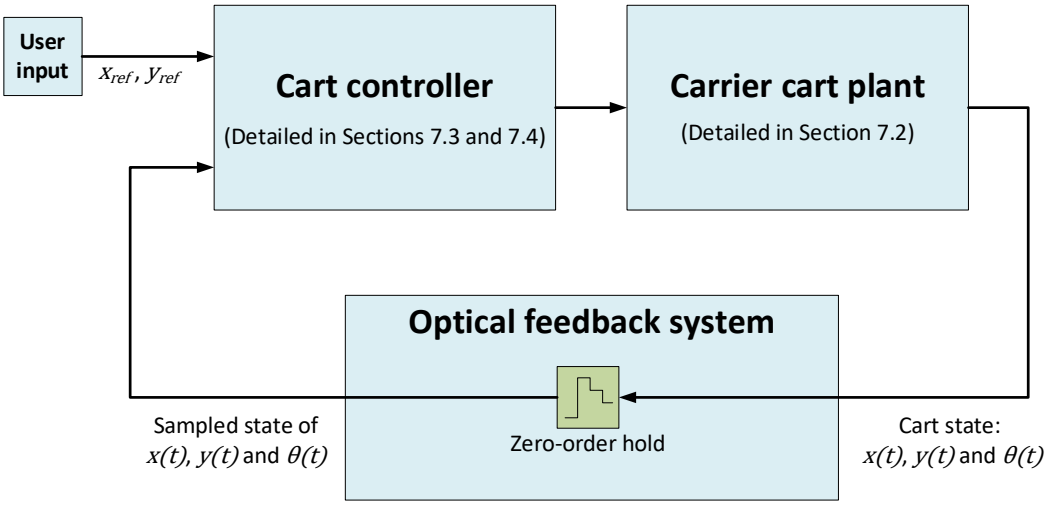


Figure 7.15 - Block diagram providing an overview of the Simulink implementation of the cart control simulation.

7.5.2 Simulation results

Figure 7.16 shows the response of the cart controller to a step input with target coordinates (x, y) = (1, 1.5) m. The initial state of the cart is (x, y, θ) = (0,0,0). The cart reaches the target coordinates within the desired steady state error, with the x error as 0.021 m and the y error as 0.041 m. The approach in the $\bar{y}$ direction has no overshoot, while the approach in the $\bar{x}$ direction has an overshoot of 2%. With regards to propellant use, this manoeuvre used up 5.36 s of thrust time. All design requirements are thus met for the simulated cart controller.

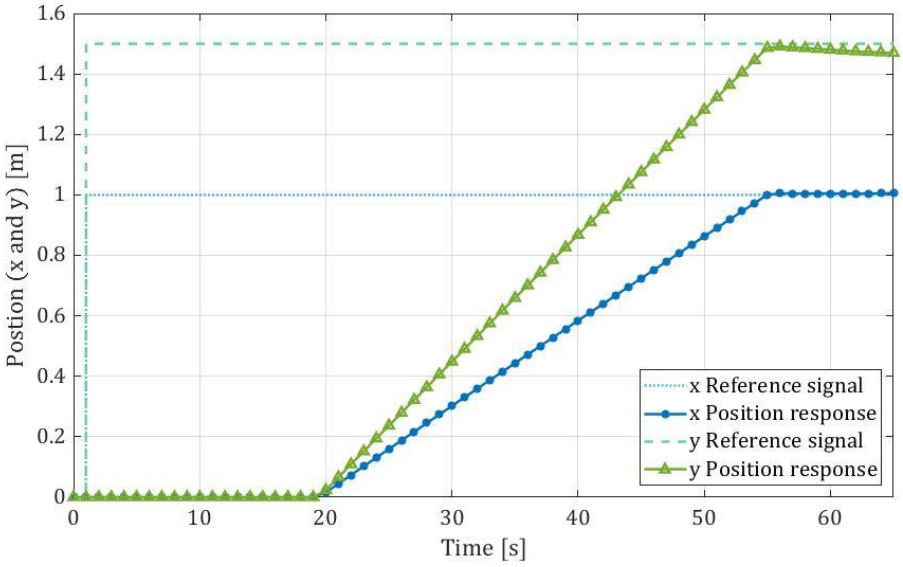


Figure 7.16 - Plot showing the step response of the cart controller to the target coordinates of (x, y) = (1, 1.5) m.

The reference tracking of each of the low-level controllers during this manoeuvre is shown in Figure 7.17 and Figure 7.18. Both controllers track their respective references in compliance with their designed specifications discussed in Section 7.4. It is noteworthy from Figure 7.17 that the reference input for the heading angle in the case of a cart manoeuvre is not simply a step input, but slowly changes as the cart approaches the target coordinates. This is because the angular position controller must make corrections for inaccuracies that occurred earlier during the control. Despite this, the angular position controller still tracks the reference signal well.

When examined together, Figure 7.17 and Figure 7.18 give a good indication of the functioning of the reference calculator subcomponent of the cart controller (discussed in Section 7.3.1). From Figure 7.18, the start of the velocity profile occurs at around 19 s, which is t_θ in this case. This is the point in time where the reference calculator considers the angular position as 'settled'. Similarly, t_p is seen to be around 55 s, where the cart reaches the proximity of the target coordinates. At this point the cart is commanded to slow down to zero velocity.

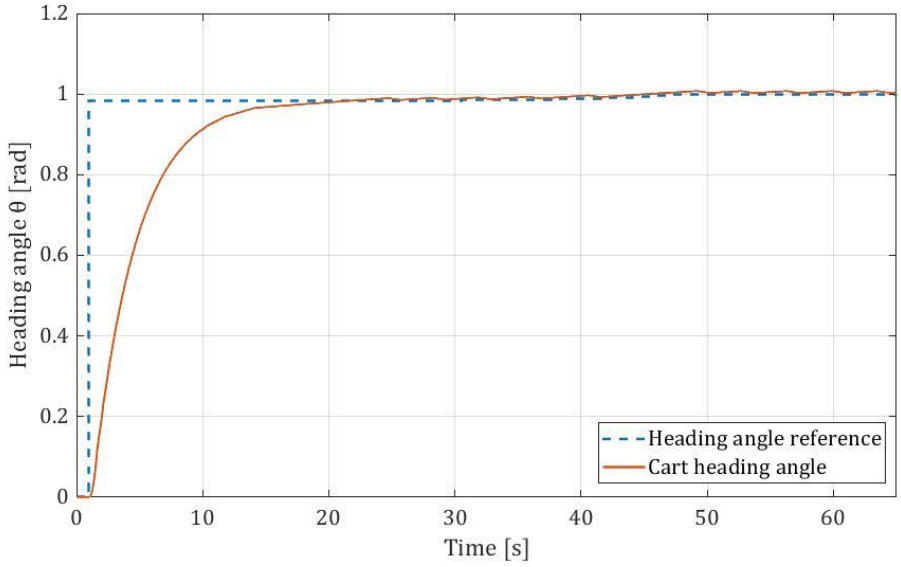


Figure 7.17 - Reference tracking of the angular position controller during the cart manoeuvre.

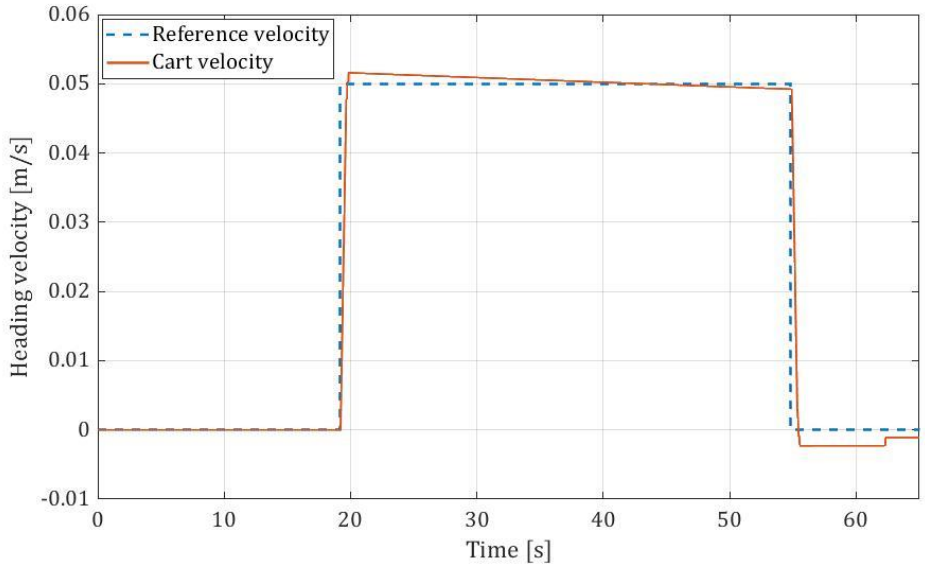


Figure 7.18 - Reference tracking of the heading velocity controller during the cart manoeuvre.

The effect of noise was also investigated in a separate simulation – but for the levels of noise seen from initial tests with the FCS's optical pose estimator, the simulated controller was successful and had similar results compared to the case

above. The frequency of the noise seen in the optical pose estimator's measurements was 5 Hz – this is significantly higher than the bandwidth of the two low-level controllers, which are both at around 0.3 Hz. Since the low-level controllers' bandwidths are so much lower than the noise frequency, the cart controller was still successful with the inclusion of noise in the model. If noise is seen to be a bigger problem in the practical implementation, more measures will be taken to mitigate the effect thereof.

7.6 Hardware-in-loop control simulations

7.6.1 Hardware-in-loop simulation overview

The designed cart controller performs acceptably in the pure simulation case. This section looks at simulating the cart controller with hardware in the loop, using the microcontroller that is implemented on the practical system. As discussed in Chapter 4, the demonstration payload's microcontroller is the ESP8266, which has built-in Wi-Fi capabilities. To try and emulate the practical system as best as possible in the hardware-in-loop (HIL) tests, the Robot Operating System (ROS) is used as the communication medium between the desktop computer running the Simulink model and the ESP8266 running the cart controller.

Figure 7.19 shows the block diagram overviewing the Simulink model used to run the HIL simulations of the cart controller. The model is similar to that used for the pure simulation case (shown in Figure 7.15), with the key difference being the ROS communication blocks, and the fact that the cart controller is run by the ESP8266 instead of in Simulink itself. The ESP8266 communicates with the Simulink model over Wi-Fi using ROS as an interface. The ROS Toolbox for MATLAB is used for the ROS communication blocks on the Simulink model side.

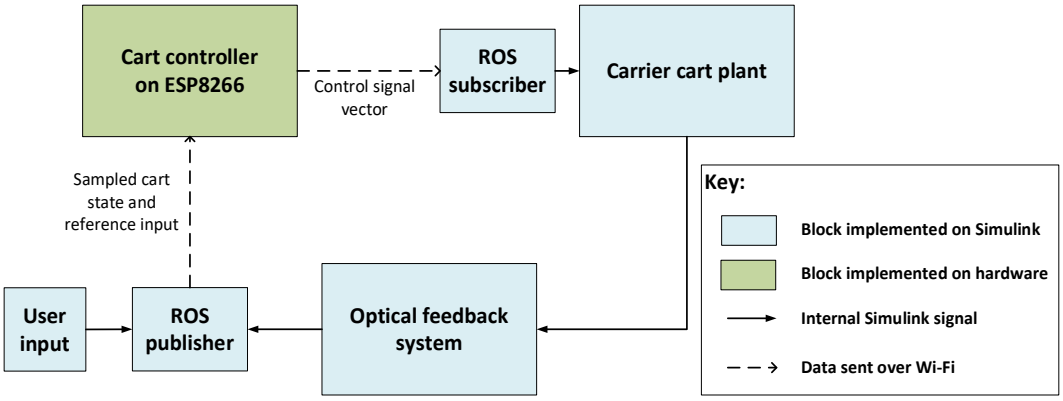


Figure 7.19 - Block diagram showing the structure of the Simulink model used to run the HIL simulations.

7.6.2 Microcontroller implementation of the cart controller

The two low-level continuous domain controllers designed for the cart controller need to be discretised and converted into difference equations in order to be implemented on a microcontroller. Two discretisation methods were considered, Tustin's method and the pole-zero mapping method. For this case, both were found to have negligible differences, so either could be used. Both discretised controllers have the general form of

$$D(z) = \frac{a_1 z + a_2}{b_1 z + b_2}, \tag{7.22}$$

Where a_1, a_2, b_1 and b_2 are tied to the controller parameters. This discretised controller converts to the difference equation

$$u(k) = \frac{1}{b_1}[a_1 e_c(k) + a_2 e_c(k-1) - b_2 u(k-1)]. \tag{7.23}$$

In the difference equation, $u(k)$ is the control output value at the current timestep k, $e_c(k)$ is the error value at the current timestep, and $e_c(k-1)$ and $u(k-1)$ are the error value and control output value at the previous timestep. This is the equation form used by the microcontroller to determine the required control output.

A control rate of 10 Hz was chosen to be compatible with the capabilities of both the solenoid valve and the optical pose estimation system. As detailed in Chapter 6, the microcontroller makes use of the ROS Serial library to communicate with the rest of the system.

7.6.3 Simulation results

Figure 7.20 shows the simulated HIL step response of the cart controller as implemented on the ESP8266. The target coordinates are (x, y) = (1, 1.5) m as before. The target coordinates are successfully reached by the cart within the steady state error requirements, the x error being 0.048 m and the y error being 0.029 m. There is no overshoot on either of the approach directions. The total thrust time is 14.2 s, which is still within the desired thrust time specification.

The HIL simulation results are comparable with the pure simulation results. The magnitude of the steady state error in the $\bar{x}$ and $\bar{y}$ directions are similar, and so is the overshoot of each approach. The main difference seen is the propellant use, where the HIL simulation results predict a higher thrust time and thus a higher propellant use. The HIL simulation uses almost 3 times as much propellant. This is likely caused by the HIL simulation having to make more corrections, as practical implications such as communication latency start to play a role. All the

specifications for the cart controller are met in the HIL simulation of the microcontroller implementation.

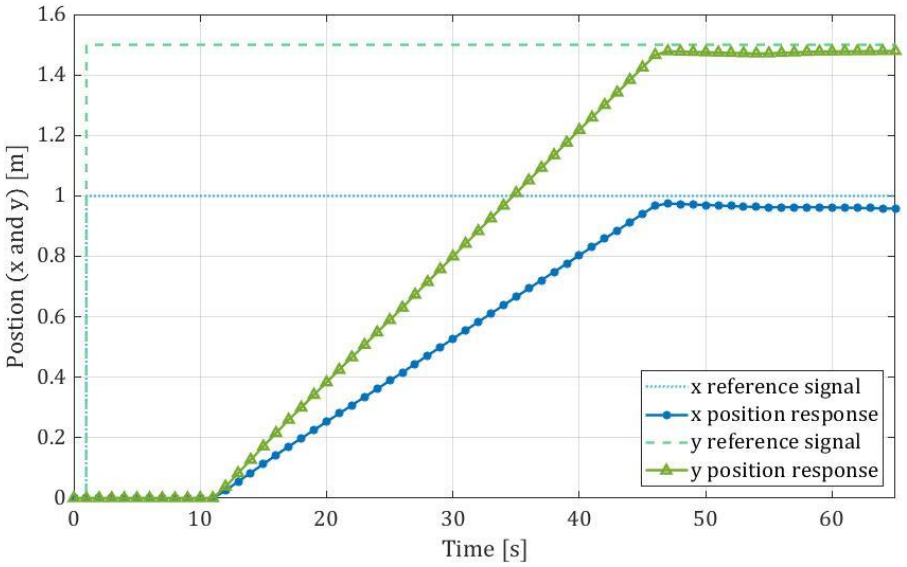


Figure 7.20 - Simulated HIL step response of the hardware implementation of the cart controller to the target coordinates of (x, y) = (1, 1.5) m.

Figure 7.21 and Figure 7.22 show the reference tracking of the low-level controllers during the HIL simulation of the cart manoeuvre. The hardware implementation of the angular position controller performs good reference tracking and is seen to rise and settle faster than the simulated continuous case. The faster response is attributed to the use of a discretised version of the controller. The hardware implementation of heading velocity controller is seen to have more overshoot and a higher steady state error than its simulated continuous counterpart. This is also attributed to the discretisation of the controller.

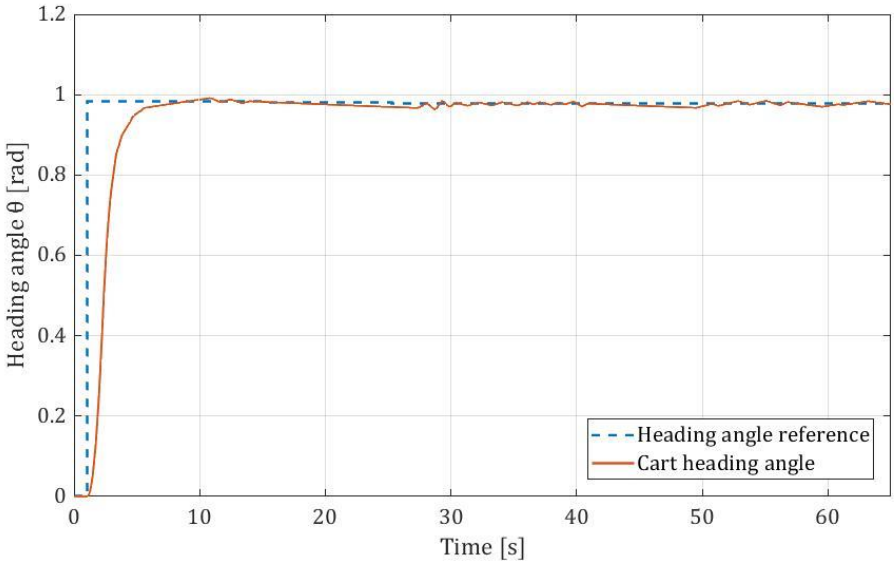


Figure 7.21 - Reference tracking of the angular position controller during the HIL simulation of the hardware implementation.

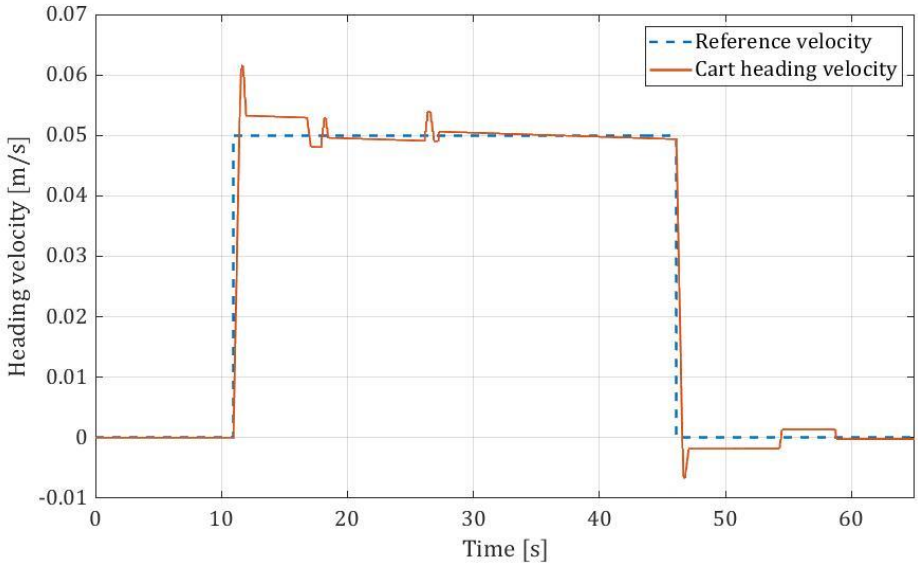


Figure 7.22 - Reference tracking of the heading velocity controller during the HIL simulation of the hardware implementation.

7.6.4 Implementation issue - Jitter

The results shown in the previous section are for a single HIL simulation run, nevertheless the simulation was repeated multiple times to observe how consistent the results were. For the most part, the majority of simulation runs

showed small variations. However, some anomalies were encountered where the simulation showed significantly different results.

The variations seen were most obvious when looking at the simulated HIL response of the angular position controller, mostly at the beginning of the manoeuvre. Figure 7.23 shows the angular position controller responses from several HIL simulation runs. All runs start with the same initial conditions and have the same inputs. It is evident that there is significant variation in the case of some runs, namely run 1 and run 6 in Figure 7.23.

The suspected cause is the jitter in the latency of the Wi-Fi connection. In this case, the jitter refers to the variation in latency of the Wi-Fi connection – which causes delays in both the sending and receiving of messages to and from the ESP8266. Jitter was seen to be an issue when initially developing the software on the ESP8266, with the latency sometimes showing variations as large as 500 ms in some cases. This equates to five control cycles, which is significant. The average latency was seen to be around 15 ms. To confirm that the jitter is the cause, additions were made to the Simulink model used for the pure simulation of the cart controller (discussed in Section 7.5). The added feature is the inclusion of an instance of jitter in the model, where the jitter duration and the time at which the jitter occurs could be altered. This was used to simulate the effects of jitter on the cart controller.

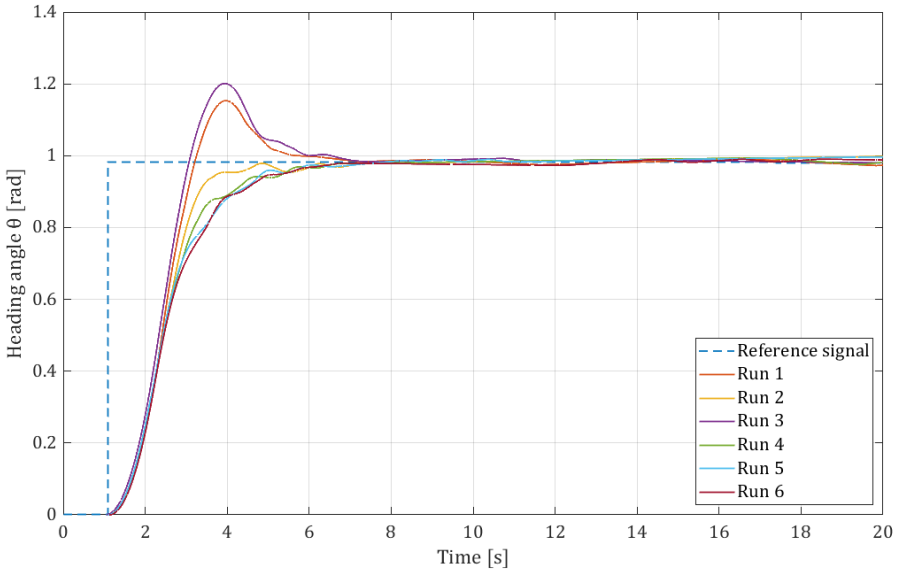


Figure 7.23 - Angular position controller responses from several runs of the HIL simulation, showing the variations between runs with the same initial conditions and inputs.

Figure 7.24 shows the angular position controller responses from several pure simulations of the cart controller with the jitter included at the 2 second mark with increasing jitter durations. The base case for the discrete controller with no jitter is also shown. It is evident that overshoot increases with increased jitter duration, and that higher jitter durations induce similar responses to the anomalies seen in Figure 7.23. This gives good confidence that jitter is the cause for the significant variations seen in multiple HIL simulations.

In order to mitigate the effect of jitter, a basic estimator is implemented on the cart controller on the ESP8266. The form of the estimator is simply

$$\theta_{estimated}(k) = \theta(k-1) + [\theta(k-1) - \theta(k-2)]T_s, \qquad (7.24)$$

where $\theta(k-1)$ is the sampled θ value one timestep ago, $\theta(k-2)$ is the sampled θ value two timesteps ago, and T_s is the controller sample time. The estimated value of theta is only used if the microcontroller doesn't receive a message from the optical pose feedback system within the sample time, i.e. if the jitter duration is longer than 100 ms. The same estimation is accordingly applied to the heading velocity.

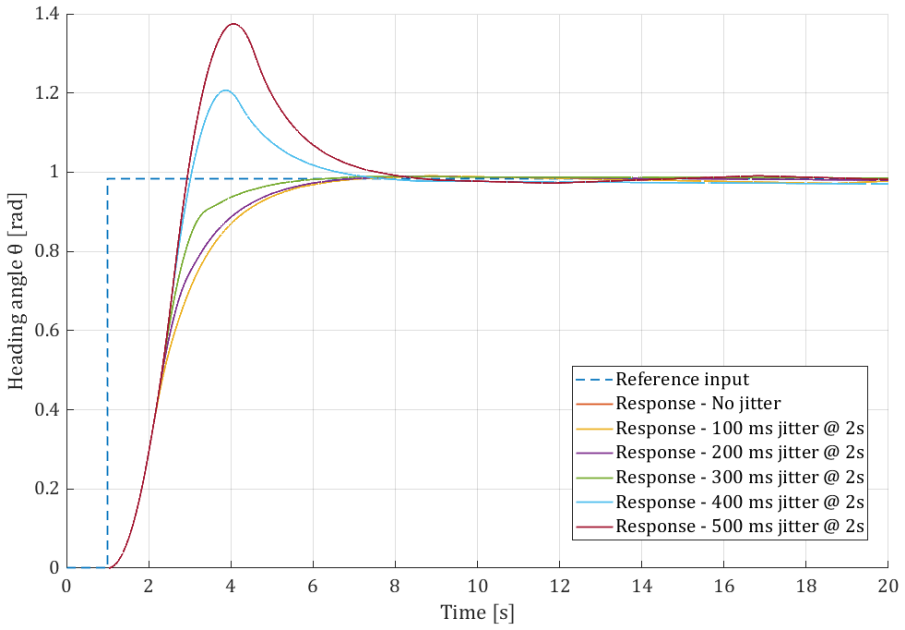


Figure 7.24 - Plot showing responses of the discrete angular position controller from several pure simulations of the cart controller, with varying amounts of jitter included. Jitter occurs at the 2 s mark for each run, with the jitter duration increasing from 0 to 500 ms between runs.

Figure 7.25 shows the effectiveness of the jitter mitigation strategy. Several runs of the HIL simulation are once again shown, but now with the estimator implemented on the cart. The estimator evidently assists in lowering the variation between runs.

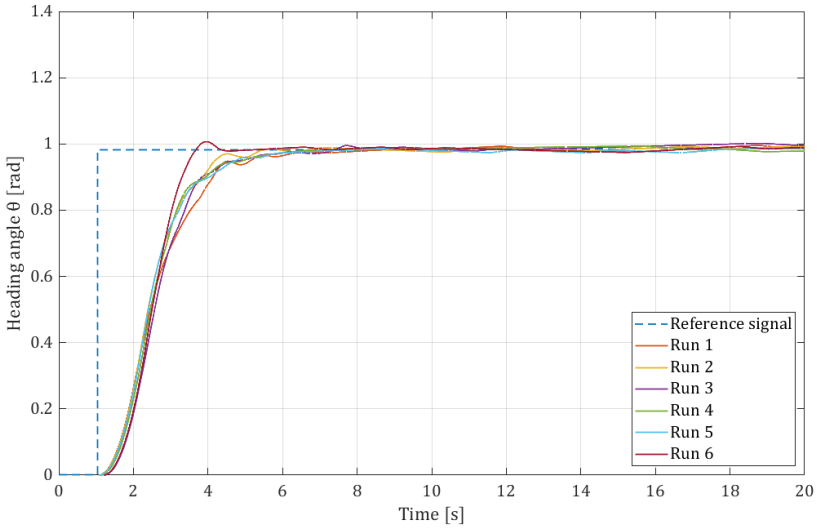


Figure 7.25 - Angular position controller responses from several runs of the HIL simulation, using the estimation for the jitter effect mitigation.

7.7 Summary and conclusion

This chapter discussed the modelling, design, and simulation of the cart control system in the continuous domain. The designed cart controller was also discretised and implemented on a microcontroller, which was used for hardware-in-loop simulations and will be used for the practical implementation of the cart controller. The cart controller can successfully perform a basic step manoeuvre in simulation. The hardware-in-loop simulation resembles the pure simulation case well, aside from some practical implications. One practical implication was that the jitter present on the Wi-Fi connection caused significant variations in several control runs with the same inputs. The jitter is mitigated using a basic estimator on the controller, which is used only when no messages are received over Wi-Fi in an expected timeframe. The jitter issue can likely be improved by setting up a direct Wi-Fi connection between the FCS's desktop computer and the payloads instead of relying on a Wi-Fi router.

With the demonstration payload's control system now successfully implemented, the test facility can be demonstrated. The following chapter discusses the practical implementation and demonstration of the test facility and the results thereof.

8 Practical implementation and results

This chapter discusses the practical implementation of the test facility and the results related to the system's performance and functioning. Furthermore, this chapter highlights practical problems faced during implementation and discusses the implemented solutions for each problem solved and recommends solutions for unsolved problems.

8.1 Performance of the carrier cart

8.1.1 Measurement of design specifications

The final implementation of the filled carrier cart with the payload and the ArUco marker mounted has a measured mass of 8.1 kg and a size of 338x330x297 mm. This meets the specifications set up for the size and mass of the carrier cart.

To determine the run time of the cart, a simple script was written for the demonstration payload which turns on a thruster pair for 2.5 s, then turns on the opposing pair of thrusters for 2.5 s, and repeats. The cart was filled to its design pressure of 124 bar, placed on the table and allowed to hover. The script was then executed, and the amount of cycles of thrust that the cart performed were counted until the gas receiver's pressure reached 10 bar, at which point it is considered empty in the design. The amount of thrust time that the cart achieved was measured as 90 s. This infers that the run time that the carrier cart achieves, with a single thruster duty cycle of 10%, when at full pressure is approximately 15 minutes. This translates to an estimated delta-v value of 4.4 m/s. This meets the specification of more than 10 minutes of run time at 10% single thruster duty.

It is noteworthy that the thrust time achieved is about 10% higher than calculated for the nozzles used on the carrier cart. For the 0.4 N nozzles used on the cart, with a measured specific impulse of 27.5 s, the expected thrust time was calculated as 81.2 s – but 90 s was measured. The reason for this discrepancy is speculated to be caused by the increased density of the nitrogen stored in the cart's tank in practice. To confirm this, some calculations were done to determine the approximate temperature at which the cart's tank is filled. The design assumes that the cart's tank is filled with nitrogen at 20°C - this was chosen as the design temperature since the cart could be left filled overnight and would settle to this temperature. The temperature drop across the fill system's regulator is estimated as 22°C. This implies that if the industrial cylinder is at 20°C when filling, the cart's tank will be filled with nitrogen at -2°C. The density of nitrogen at -2°C is 8.4% higher compared to its density at 20°C (as designed for), which is indeed within proximity to the 10% discrepancy of the thrust time. Practically, this means that the cart will achieve a slightly longer run time if used shortly after filling it. If left

to settle to room temperature, the cart's tank pressure will rise but will be relieved by the relief valve if it exceeds safe storage pressures.

The carrier cart performs its functions acceptably well. It is able to support the payload with a low friction interface on the table and the thrusters successfully move the cart as expected with an acceptable run time.

8.1.2 Practical issues observed

The practical issues faced with the carrier cart's practical implementation included the levelling of the glass platform, interference of particulates on the table, and slight variations in working pressure. The following subsections discuss these problems, the practical implications thereof, and possible solutions.

8.1.2.1 Levelling of the glass platform

The levelling of the glass platform is crucial for the test facility's function of emulating the micro-gravity environment (as also discussed in Chapter 2). In this context, the levelness of the surface refers to the surface's perpendicularity to the gravity vector. If the platform is not level, the carrier cart tends to drift into the valleys of the table because of the low friction interface. The test facility's glass table makes use of turnbuckles with open hooks to apply tension to its structural supports, thus allowing for a levelling mechanism for the glass platform. A single turnbuckle of the mechanism is shown in Figure 8.1. These turnbuckles are positioned in an array around the base of the table, allowing for the height of points on the table to be slightly adjusted by tensioning the cables attached to them.

Figure 8.1 - Photograph of a single turnbuckle which forms part of the glass table's levelling mechanism.

The levelling process is rather tedious without specialised tools. The table was levelled by using the carrier cart hovering on the table to map the valleys, and then adjusting the turnbuckles until the area around the valley was found to be approximately level. This method provided a reasonably level surface but can be improved upon – for example by use of small laser levels or small machinist levels. However, these types of levels are typically expensive. The table was observed to slowly go out of level over time and will likely need adjustment every few months.

8.1.2.2 Interference caused by particulates

Another problem faced by the carrier cart is the interference caused by small particulates. These particulates are introduced by dust and dirt build-up on the table over time. When the carrier cart's air bearing hovers over an area with a large amount of dust on the table it causes interference with the air bearing, and the friction is momentarily increased by a large amount. This problem can be solved easily by thoroughly cleaning the glass platform before experiments and covering up the table with a sheet when not in use.

8.1.2.3 Variations in working pressure during operation

During operation the carrier cart's set working pressure does not stay perfectly constant. This occurs due to a nonideality of pressure regulators known as droop – which causes the outlet pressure of a regulator to change based on flow rates and input pressure. This behaviour was accounted for as far as reasonably possible in the design phase but is mentioned here because of its practical implications. When set to 4.5 bar initially, the working pressure was observed to vary between 4.7 bar and 4.3 bar during operation. The implication of this is that thrust levels can vary by ±5% during operation. The effect of this is not very significant but is something that future users need to be aware of in cases when an experiment is sensitive to thrust levels. If necessary, this variation can be decreased by using nozzles with a lower force output (and thus a lower flow rate).

8.2 Performance of the feedback and communication system

8.2.1 Measurement of design specifications

To determine the accuracy of the translational position measurement, the cart was placed at 15 different positions on the table. The positions were spread out to cover as much of the table as possible. The x and y position of the cart at each location was then measured and recorded using a measuring tape with reference to the origin marker. Additionally, the x and y position of the cart at each location was measured and recorded using the feedback and communication system (FCS). The data measured using the measuring tape was taken as the ground truth, and

the FCS's was compared to this data. The average error was seen to be 1.2 cm, while the maximum error seen was 1.9 cm. From this it is inferred that the FCS meets the requirement for an accuracy within 3 cm.

A similar process was followed for determining the accuracy of the rotational position measurement. For this test, the carrier cart's marker was placed at different locations on the glass platform with different orientations. The ground truth measurements were taken using a machinist's protractor and a straight edge to reference the table's edges. The angle between the origin marker and the table's edges was also measured for this test. Comparing the orientation measured by the FCS to the ground truth values, an average error of 0.04 rad and a maximum error of 0.06 rad was recorded. From this result it is inferred that the FCS meets the rotational position accuracy requirement of less than 0.1 rad error.

For the velocity and rotation rate noise level requirements of the FCS, it was found that they were initially not met using the raw measurements from the FCS's optical estimator. For this reason, a digital filter was added to the practical implementation of the FCS.

The filter was designed in the continuous domain, then converted to a digital filter and implemented using difference equations similarly to the demonstration payload's low-level controllers. The frequency of the noise was observed to be around 5 Hz, which makes sense since it is the Nyquist frequency of the optical pose estimator. A lowpass filter is used with a break frequency of 1.5 Hz to attenuate the noise. The break frequency was chosen to be high enough such that the demonstration payload's controllers would not be significantly affected. The filter was included in the simulation of the control system to confirm that the controller was unaffected. It was recognised that this break frequency will not be suitable for all experiments, so the implementation of the filter in the FCS includes an option for users to change the break frequency. Additionally, the FCS outputs both the filtered and unfiltered pose estimation data. The filter is implemented as part of the *Cart pose publisher* ROS node which forms part of the optical pose estimator subsystem of the FCS. The filter is applied to the x, y and θ measurements.

Figure 8.2 compares the filtered and unfiltered heading velocity measurements taken by the FCS while the cart was stationary. The magnitude of the noise is significantly reduced by the filter and is brought within the specifications set up for the FCS for the heading velocity.

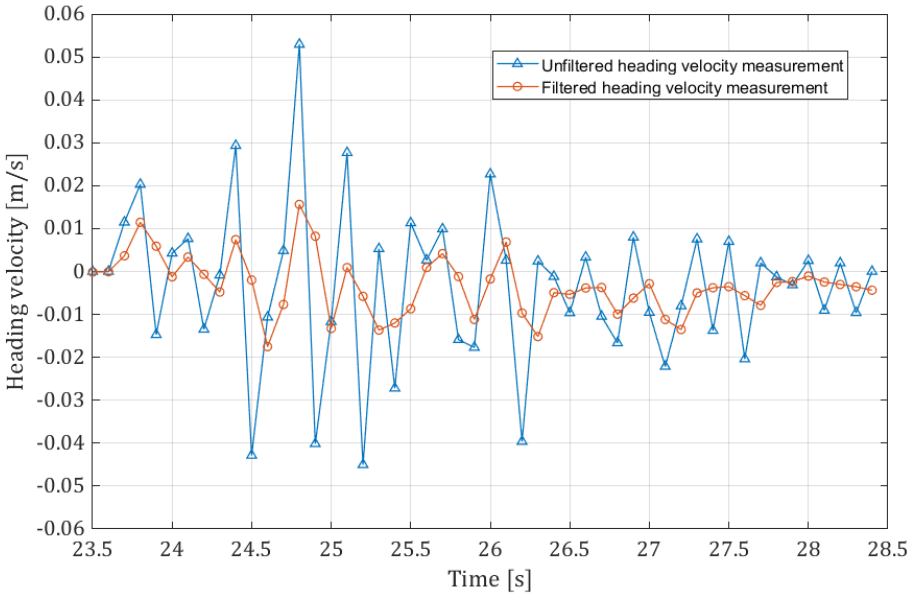


Figure 8.2 - A plot comparing the filtered and unfiltered measurements of the cart's heading velocity when stationary.

To measure the noise specifications of the FCS, the cart was placed at different locations on the table and kept stationary for the measurements. The noise was seen to increase as the marker was moved further from the camera and also as the marker moved closer to the edges of the image (which is attributed to distortion). The maximum noise was measured as 0.04 m/s for the heading velocity and 0.05 rad/s for the rotational velocity. This meets the required specifications of less than 0.05 m/s and 0.1 rad/s respectively.

The FCS performs its functions acceptably well. It is able to measure the position and velocities of multiple carrier carts (this was tested by using multiple markers) and can communicate this information to multiple cart payloads. Communication with multiple payloads was tested using hardware in loop simulations (since only one physical carrier cart was built at the time of testing) and was successful. The measurements are also successfully displayed to the user and can be logged.

8.2.2 Practical issues observed

Some practical issues observed with the functioning of the FCS include the reliability of marker detection, noise increases as the carts move further from the camera, and occasional inversion of the marker's estimated pose. The following subsections will detail these problems and possible solutions.

8.2.2.1 Reliability of marker detection

Initially it was found that marker detection was not very reliable, and the pose detection algorithm would have trouble detecting the markers in certain positions on the table. Some troubleshooting revealed that the cause of this issue was the amount of whitespace around the marker. Increasing the whitespace around the marker increased the detection reliability significantly. In its current state, the optical pose estimator does still occasionally lose track of a marker momentarily, but this was seen to happen closer to the far edge of the table and only occurred for one or two sample periods at a time. When the optical pose estimator loses track of the marker momentarily, the FCS communicates the last known pose of the marker. The effect of this was seen to be insignificant, at least in the case of the demonstration payload's control. The detection reliability can likely be further improved by tuning camera parameters such as gain and whitebalance, and by experimenting with mounting the camera in different locations for more perpendicular viewing angles. Additionally, an estimator can be implemented to estimate the pose of the cart based on its previous states when marker detection is lost momentarily.

8.2.2.2 Increased noise further from the camera

It was observed that the optical pose estimator's measurements exhibit more noise as the measured marker is moved further from the camera. This also occurs when the marker was moved closer to the edges of the table where lens distortion is more severe. However, as mentioned in Section 8.2.1, the noise levels measured were always below 0.05 m/s and 0.1 rad/s for heading velocity and rotational velocity respectively (when stationary) – which is within the required specifications. It is speculated that the increase in noise is caused by the fact that the pose estimation algorithm has less pixels per metre to work with and more distortion to deal with in the problem areas on the table. The noise levels can likely be reduced and made more uniform by mounting the camera closer and more perpendicular to the table.

8.2.2.3 Occasional pose inversion (marker pose ambiguity problem)

Another issue observed with the optical pose estimator was the occasional inversion of the normal axis of a detected marker's pose. This was seen to occur more often near the edges of the table, so lens distortion seems to amplify this problem. The effect of this inversion is that it causes a spike in the measurement of the cart's heading angle, normally seen to be around 0.2 rad in magnitude. An example of this can be seen at around the 18 second mark in in Figure 8.4 which will be discussed in the following section. For the most part this problem did not occur often closer to the middle of the table and did not have a significant effect on the control of the demonstration payload.

Upon further investigation, this issue was found to be linked to a problem referred to in literature as the marker pose ambiguity problem (Ch'ng et al., 2020). This issue is an inherent problem with using planar markers to estimate pose, because in some orientations there can be more than one possible solution. The issue is generally associated with noise in measurements and can thus possibly be mitigated by decreasing noise levels in the FCS's measurements. Additionally, the problem can be mitigated by using multiple markers per cart and implement a voting system where the algorithm uses the pose of multiple markers to determine which orientation is most probable. A grid of 4x4 markers with a side length of 5 cm was tested on the carrier cart, and the markers were successfully detected by the optical pose estimator – so the software can be expanded to use multiple smaller markers with a voting system. Furthermore, various solutions to this problem exist in the literature including the work done by Ch'ng et al. (2020) which is based on rotation averaging, as well as the method proposed by Jin et al. (2017) which is based on sensor fusion.

8.3 Practical demonstration of the test facility using the demonstration payload

This section presents and discusses the results from a control manoeuvre performed by the demonstration payload. The intent of this control manoeuvre is to demonstrate the capabilities of the test facility in practice.

Prior to testing the full cart controller, the low-level rotational position controller and heading velocity controller from Chapter 7 were tested individually. The heading velocity controller was found to perform acceptably, while the rotational controller required some tuning since it was showing significant oscillations in its response in practice. The low-level controller parameters used for the practical tests are listed in Appendix D.1.

Due to practical implications such as sensor noise and the practical performance of the low-level controllers, the cart controller's reference calculator parameters (discussed in Subsection 7.3.1) also had to be tuned. The final values used for the demonstration control manoeuvre are listed in Table 8.1. These values can be tuned further to achieve better performance.

Table 8.1 - Tuned values of the cart controller's reference calculator parameters used for the demonstration control manoeuvre.

Control parameter	Final chosen value for simulations
v_{max}	0.15 m/s
θ_{thresh}	0.08 rad
Δ_{thresh}	0.015 rad/s
d_{thresh}	0.2 m

The demonstration payload's cart controller was tested in practice with several reference coordinates and initial conditions, with success in most tests after some tuning. The test data presented here was chosen because it demonstrates the functioning of the test facility well, while also highlighting some of the practical issues discussed in Section 8.1 and Section 8.2.

Figure 8.3 plots the x and y position of the cart along with the reference signals of each versus time during the demonstration control manoeuvre. The initial position of the cart is (x, y) = (1.02, 0.4) m and a reference command with the coordinates (x_{ref}, y_{ref}) = (1.5, 1) m is sent to the cart controller at t = 1 s. The cart successfully moves to the reference coordinates within a steady state error of 2 cm in the $\bar{y}$ direction and 12 cm in the $\bar{x}$ direction. Because of the large steady state error in the $\bar{x}$ direction, the cart controller unfortunately does not meet the requirements initially set up in Chapter 7. Some more tuning will likely yield the required results and the features of the test facility's FCS do ease the tuning process by providing wireless parameter setting during operation. However, the results shown by this this manoeuvre are good, and it successfully demonstrates the test facility's capabilities.

It is noteworthy that at t = 11.6 s, the optical pose estimator lost track of the marker for one sample period. This is the reason for the two consecutive samples that have the same value. As discussed in Subsection 8.2.2.1, this was found to have no significant impact on the control in the tests performed.

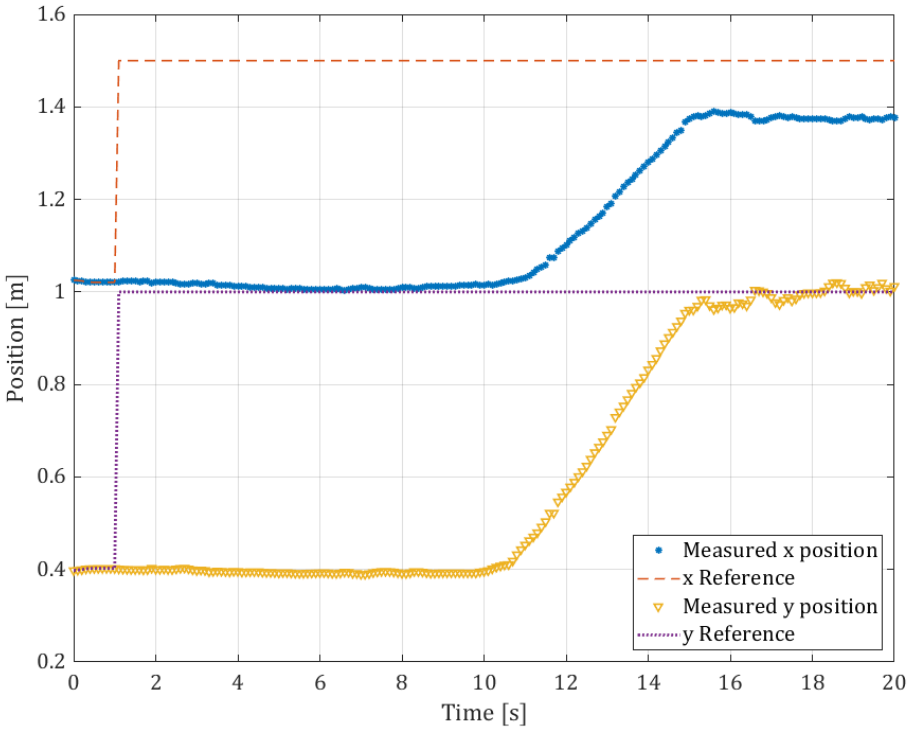


Figure 8.3 - A plot showing the x and y position of the cart versus time during the demonstration control manoeuvre. Also shown are the reference signals.

Figure 8.4 plots the response of the cart's low-level rotational controller during the demonstration manoeuvre. Compared to the hardware in loop (HIL) simulation results presented in Chapter 7, the rotational controller implemented in practice has a slower response and a larger steady state error of about 0.1 rad. The main reason for this is that the controller's gain was decreased during the tuning process to eliminate oscillations in the response. Further tuning of this low-level controller will yield better results both for itself and the high-level cart controller – and is recommended if the demonstration payload is to be used in the future.

At t = 18.3 s in the plot, the effects of the momentary normal axis inversion are seen (caused by the marker pose ambiguity problem discussed in Subsection 8.2.2.3). At this point in the test, the optical pose estimator could not resolve the ambiguity of the marker's orientation and thus a spike in the heading angle measurement occurred. The control was not significantly affected by this issue, aside from a waste of control energy – fortunately the lower bandwidth of the controller makes it less sensitive to this spike in the sensor feedback.

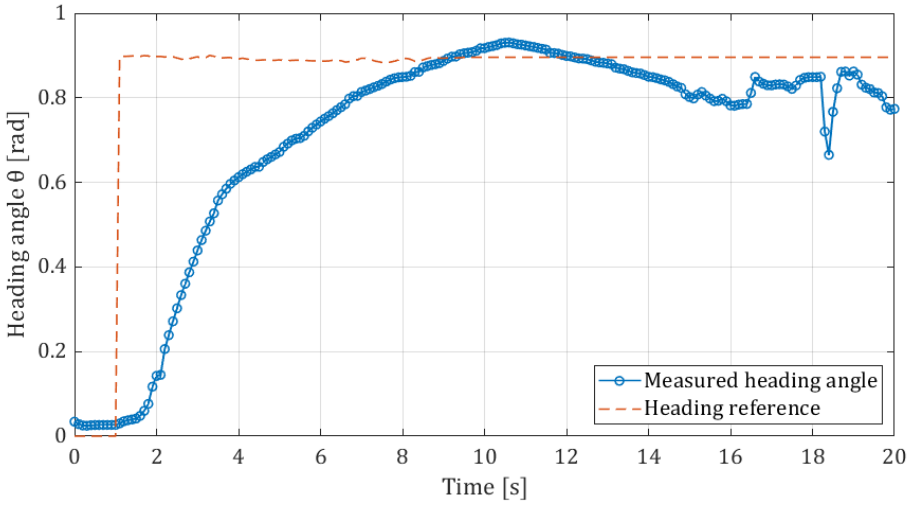


Figure 8.4 - A plot of the cart's heading angle versus time during the demonstration control manoeuvre.

Figure 8.5 plots the torque command (control signal) output from the rotational controller during the demonstration manoeuvre. This data was logged by using the FCS's bi-directional communication feature to receive control output from the payload. Note that the actuators are saturated at a torque magnitude of 36 mN but are not saturated for a significant portion of time during the test. Most of the control energy is spent at the beginning and the end of the manoeuvre. The majority of control energy spent at the end of the manoeuvre is attributed to the normal axis inversion that occurs due to the marker pose ambiguity problem, so fixing this will yield better performance in terms of propellant usage.

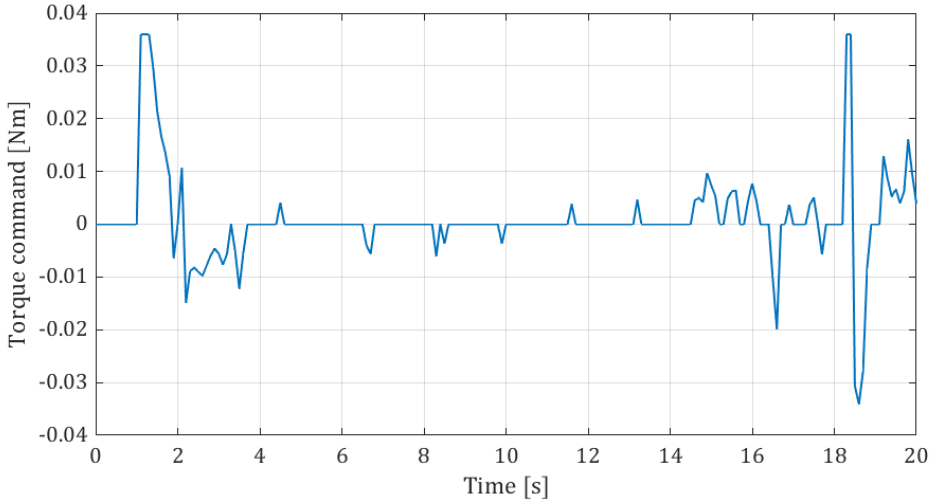


Figure 8.5 - A plot of the torque command output of the low-level rotation controller during the demonstration manoeuvre.

The response of the demonstration payload's heading velocity controller is plotted in Figure 8.6. The controller tracks the reference input reasonably well considering the noise present on the velocity measurement. The response has a rise time of 2.15 s which is comparable to that seen in the HIL tests in Chapter 7. This data also shows the increase in the noise level of velocity measurements as the marker moves further away from the camera – which was discussed in Subsection 8.2.2.2.

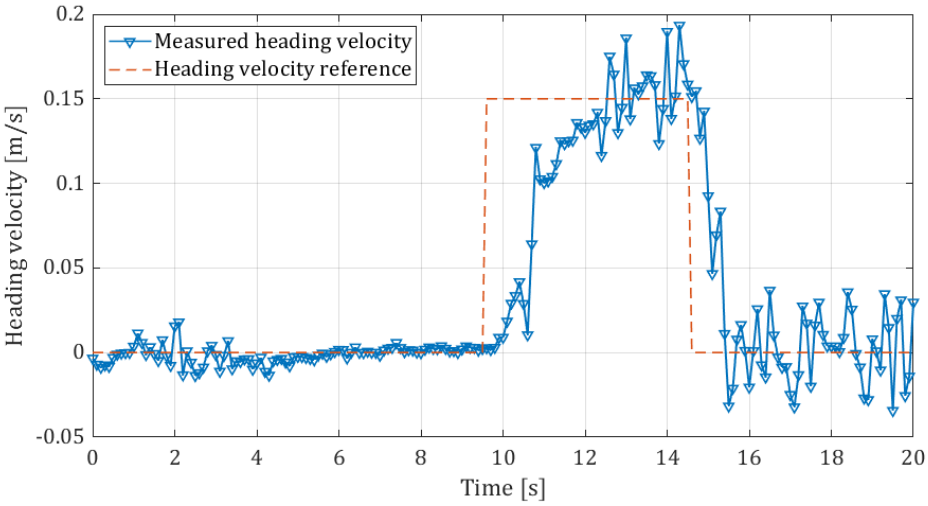


Figure 8.6 - A plot of the cart's heading velocity and reference velocity versus time during the demonstration control manoeuvre.

The force command output from the heading velocity controller versus time is plotted in Figure 8.7. Once again, this data was logged by using the FCS's bi-directional wireless communication feature. Most of the control energy is used to accelerate the cart to v_{max} and then to slow down to zero velocity again. Additionally, a smaller but significant amount of control energy is used at the end of the manoeuvre in response to the higher noise levels. Improving the controller's robustness to noise would decrease this unnecessary propellant usage.

Figure 8.8 shows the cumulative thrust usage, i.e. the total amount of time that the thrusters are activated, versus time during the control manoeuvre. This includes the thrust usage from both low-level controllers. The plot gives an indication of when propellant is expended during the experiment. From this data it inferred that roughly 70% of the propellant is expended to move the cart to the target location, while the other 30% is spent in response to practical implications seen at the end of the manoeuvre such as noise and pose ambiguity. This is evident from the fact that at t = 15.5 s, when the cart reaches the target location and comes to a stop, 7.5 s of thrust time has been expended – while at the end of the manoeuvre the total thrust time used is 10.7 s. This is a reasonable ratio of useful versus wasted propellant for the purposes of demonstrating the test facility's functioning. It can be improved by further tuning of the controller and mitigating

the effects of the practical implications mentioned. Since the total thrust time is 10.7 s, at full capacity the carrier cart is capable of providing enough propellant to perform 8 manoeuvres similar to this one.

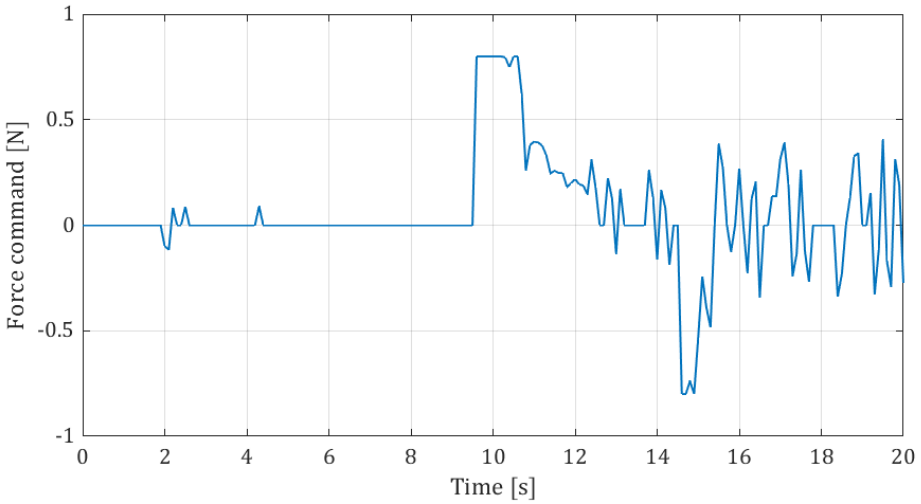


Figure 8.7 - A plot of the force command output by the low-level heading velocity controller during the demonstration manoeuvre.

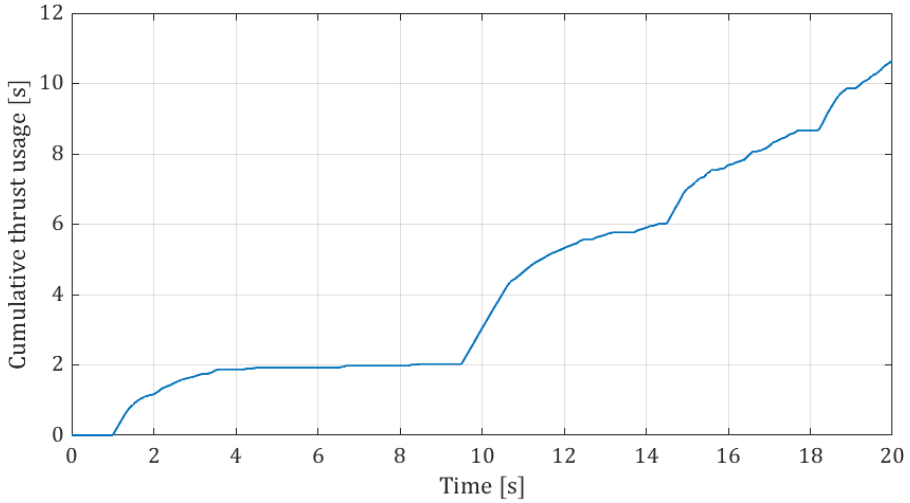


Figure 8.8 - A plot showing the cumulative thrust usage (the amount of time that the thrusters are activated) versus time during the demonstration control manoeuvre.

Overall, the demonstration payload's cart controller definitely has room for improvement in terms of steady state performance and robustness to noise. However, for the purposes of demonstrating the functioning and capabilities of the test facility, it performed well. With further tuning, the cart controller should be capable of meeting the requirements set up in Chapter 7. The key parameters

expected to impact the performance the most and thus in need of further tuning are the θ_{thresh} and d_{thresh} parameters – followed by the low-level rotational controller's parameters. Further tuning of the controller is made easier using the features of the FCS such as the wireless parameter setting functionality.

8.4 Summary, conclusion, and recommendations

This chapter discussed the practical implementation of the test facility and presented the practical results of a control manoeuvre performed by the demonstration payload. The carrier cart was found to meet all its requirements, and successful solutions were tested and recommend for the practical issues faced with its implementation. The feedback and communication system (FCS) also met all its requirements but had some practical issues that can affect some experiments that make use of the full area of the glass platform. Solutions for these problems were suggested but not all the solutions were fully tested. The demonstration payload's cart controller successfully performed a demonstration manoeuvre. While the payload's cart controller has room for improvement, it was deemed to be acceptable for the purposes of demonstrating the capabilities and functioning of the satellite test facility.

The practical implementation of the satellite test facility performs acceptably well. The specifications of the carrier cart and FCS are met, and the test facility's desired system characteristics (as laid out in Chapter 3) are met by the practical implementation.

If the experiment allows for it, it is recommended that a smaller area of the table be marked off and that the experiment be constrained to this area. This is expected to yield better performance of the FCS. Furthermore, the camera should be mounted closer to this smaller area and with a more perpendicular viewing angle. This is expected to mitigate the effects of many of the practical implications. The levelling of the glass platform will be an easier process if the area of interest is smaller. The reliability of the marker detection, the noise levels, and the marker pose ambiguity issues faced by the FCS are also all expected to be mitigated by using a smaller area on the table and by mounting the camera closer and more perpendicular to the plane of the table. Additionally, the relatively small size of the carrier cart allows for a smaller area of the table to be effectively used. For comparison, the newly designed carrier cart takes up four times less area than the previous air bearing vehicle – which the glass table was designed and constructed for.

9 Conclusion

This thesis has solved the problem of designing and implementing a planar air bearing test facility for satellite experiments. The test facility was designed to meet the needs of the Electronic Systems Laboratory (ESL) research group at Stellenbosch University. The test facility features an air bearing vehicle (the carrier cart) which moves with very low friction atop a glass platform. This low friction movement allows the test facility to be used for emulating orbital spacecraft dynamics with three degrees of freedom – two translational degrees and one rotational degree. Additionally, the test facility features a feedback and communication system (FCS) which is capable of measuring, displaying, and logging the pose information of the carrier carts on the table to the user. The FCS is also capable of wirelessly communicating with the payloads on the carrier carts, which can be used to send pose feedback to the payloads and to log control data.

The work done in this research project included the design, construction, and implementation of a functional satellite test facility. Firstly, the test facility was designed on a system level – outlining the requirements of the facility and creating a conceptual solution. Following this, the carrier cart was designed and constructed. A supporting propellant refill station was designed and constructed to refill the gas supply of the carrier cart. The carrier cart successfully facilitates low friction movement on the test facility's glass table. A custom cold gas thruster nozzle was designed for the carrier cart, which was manufactured using fused deposition modelling (FDM) 3D printing technology. The thruster nozzles were characterised using a custom-built test rig to ensure that the desired specifications are met. The digital model of the thruster nozzle was parameterised to allow for future users to easily export 3D printable models which have the dimensions they require for their application.

Once the carrier cart was successfully implemented, the test facility's FCS was designed and implemented. The FCS was implemented using the Robot Operating System (ROS) middleware, which is a popular open-source toolset mainly used for robotics. The use of an open-source software framework improves the ability to expand the FCS's software. The FCS makes use of an optical pose estimator to measure the pose of the carrier carts on the glass table. The FCS successfully measures, logs, and displays pose data in near real time during experiments. It also successfully provides control feedback and receives control information from payloads.

Upon the complete implementation of the test facility's main subsystems – the carrier cart and FCS – a demonstration payload was designed to showcase the capabilities and functioning of the satellite test facility. This payload includes a basic control system which controls the carrier cart's cold gas thrusters to perform

a manoeuvre which moves the cart from an initial coordinate on the glass table to a target coordinate.

The test facility's carrier cart has an operation time of approximately 15 minutes at a single thruster duty cycle of 10%. This is a good result and allows a long time for experiments to be run on the test facility. The dimensions of the test facility's glass platform are 3.2x1.8 m, and the carrier cart's dimensions are 338x330x297 mm, allowing ample room for experiments. The FCS can communicate, display, and log measured pose information at 10 Hz – and provides a position measurement accuracy within 2 cm.

The practical implementation of the carrier cart and FCS revealed some minor practical issues, but overall, the test facility performed well. The carrier cart and FCS successfully met all their requirements. Practical results were recorded during a manoeuvre performed by the demonstration payload. The demonstration payload successfully performed a manoeuvre in practice, and the capabilities of the test facility were showcased with these results.

The main issues with the test facility were found to be related to the practical implementation of the FCS. The issues include sensor noise and the ambiguity of measured orientations – caused by the marker pose ambiguity problem. Both issues are fundamental problems related to the use of a single camera for optical pose estimation. Recommendations were made for these issues and these recommendations are repeated in the following section. The open-source and modular nature of the FCS's implementation will ease the future implementation of these recommendations.

9.1 Recommendations

The recommendations for improving the performance of the test facility are as follows:

- Adjust the FCS's pose estimation algorithm to use multiple markers per cart. A voting system can then be used to determine the orientation of the cart, mitigating the effects of the marker pose ambiguity problem. The software is readily expandable for this and the algorithm is capable of detecting smaller markers, and so this implementation is expected to be relatively straightforward.
- If the experiment allows for it, make use of a smaller area of the table, which is close to the optical pose estimator's camera and in the centre of its field of view. This is expected to reduce the sensor noise and reduce the likelihood of the pose ambiguity problem occurring. Additionally, it reduces the area of the table which needs to be levelled, saving time.

Furthermore, the recommendations for future work to upgrade the test facility are:

- Investigate the performance of cold gas thruster nozzles manufactured by other means. For example, 3D printing by means of stereolithography or selective laser sintering could potentially produce nozzles with improved specific impulse. This will improve the operation time of the test facility.
- The implementation of a more complex estimator, such as a Kalman filter, could potentially improve the accuracy and noise characteristics of the pose information output by the FCS.
- Expand the FCS to use two or more cameras to increase the accuracy of pose measurements and to cover a larger part of the table (or a larger table in the future) without distortion introducing problems.
- The use of alternative visual markers, such as LED patterns, can be investigated. This could yield improved measurements of the cart pose, with less noise and less issues with the ambiguity problem.
- Solutions for easier bed levelling can be investigated. Levelling tools can be procured, or a custom solution can possibly be designed. This will improve the accuracy of the bed levelling process and will reduce the time spent levelling.

9.2 Contributions

This research project has contributed a functional satellite test facility for use by the ESL research group. The test facility is capable of facilitating various experiments relating to research in space-related fields. Potential experiments include:

- Docking and rendezvous experiments.
- Formation flying (with the addition of more carrier carts).
- Tests involving free-floating manipulators and debris capture.
- Model validation and control of tethered spacecraft.
- The deployment of solar panels, antennas, or sails.
- General low friction motion tests – including general tests for robotics applications.

Additionally, this project contributed knowledge concerning the performance of atmospheric cold gas thruster nozzles manufactured using FDM 3D printing. Using 3D printing to create nozzles for use on the test facility is a valuable tool for prototyping.

The versatility and expandability of the test facility allows for much potential in the way of experiments in the fields of spacecraft research and robotics research. The test facility is a useful tool for researchers to validate their projects in practice. Furthermore, having access to a test facility which can be used for practical tests provides invaluable practical engineering experience to researchers. The test facility is also useful for demonstrating functional projects to interested parties.

References

Bradski, G., 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools.*

Brown, C.D. 2002. *Elements of Spacecraft Design*. Virginia: American Institute of Aeronautics and Astronautics Inc.

Çengel, Y.A. & Cimbala, J.M. 2014. *Fluid Mechanics: Fundamentals and Applications*. 3rd edition. Singapore: McGraw-Hill Education.

Ch'ng, S., Sogi, N., Purkait, P., Chin, T., Fukui, K. 2020. Resolving Marker Pose Ambiguity by Robust Rotation Averaging with Clique Constraints. *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France.

Dieter, G.E. & Schmidt, L.C. 2013. *Engineering Design*. 5th edition. New York: McGraw-Hill Education.

Figliola, R.S. & Beasley, D.E. 2015. *Theory and Design for Mechanical Measurements*. 6th edition. Singapore: John Wiley & Sons Pte. Ltd.

Franklin, G. F., Powell, J. D., Emami-Naeini, A., Sanjay, H. S. 2015. *Feedback control of dynamic systems.* New Jersey: Pearson.

Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F. J., and Marín-Jiménez, M. J. 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition.* 47(6), June:2280-2292.

Hamedi-Estakhrsar, M.H., Mahdavy-Moghaddam, H., Jahromi, M. 2018. *Investigation of effects of convergence and divergence half-angles on the performance of a nozzle for different operating conditions*. Journal of the Brazilian Society of Mechanical Sciences and Engineering. 40(7):353.

Jin, P., Matikainen, P., Srinivasa, S.S. 2017. Sensor fusion for fiducial tags: Highly robust pose estimation from single frame RGBD. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vancouver, Canada.

Menon, C., Busolo, S., Cocuzza, S., Aboudan, A., Bulgarelli, A., Bettanini, C., Marchesi, M., Angrilli, F. 2007. Issues and solutions for testing free-flying robots. *Acta Astronautica.* 60(12):957-965.

Micci, M.M. & Ketsdever, A.D. 2000. *Micropropulsion for Small Spacecraft*. Progress in Astronautics and Aeronautics. Volume 187. Virginia: American Institute of Aeronautics and Astronautics Inc.

New Way Air Bearings. n.d. [Online]. Available at: https://www.newwayairbearings.com/ [2019, January 24].

New Way Air Bearings. 2006. *Air Bearing Application and Design Guide*. Revision E. January 18. [Online]. Available: https://www.newwayairbearings.com/sites/default/files/new_way_application_and_design_guide_%20Rev_E_2006-01-18.pdf [2019, January 25].

OAV Air Bearings. n.d. [Online]. Available at: https://www.oavco.com/ [2019, January 28].

PI Nelson Air. n.d. [Online]. Available at: http://www.nelsonair.com/index.htm [2019, January 30].

Robot Operating System. 2010. [Online]. Available at: https://www.ros.org [2019, November 10].

ROS Rosserial Documentation. 2018. [Online]. Available at: http://wiki.ros.org/rosserial [2019, November 19].

ROS Fiducials Documentation. 2018. [Online]. Available at: http://wiki.ros.org/fiducials [2019, November 18].

ROS Tf Documentation. 2018. [Online]. Available at: http://wiki.ros.org/tf [2019, November 18].

Rybus, T. & Seweryn, K. 2016. Planar air-bearing microgravity simulators: Review of applications, existing solutions and design parameters. *Acta Astronautica.* 120:239-259.

Schwartz, J.L., Peck, M.A., Hall, C.D. 2003. Historical Review of Air-Bearing Spacecraft Simulators. *Journal of Guidance, Control, and Dynamics*. 26(3).

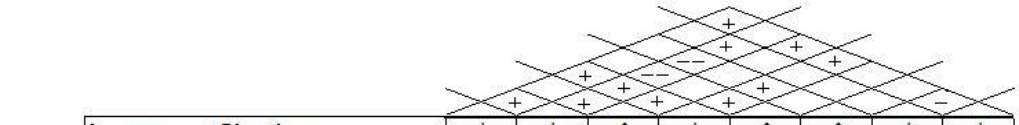
The OpenCV Reference Manual. 2018. [Online]. Available: https://docs.opencv.org/4.0.0/ [2019, November 4].

Wilde, M., Clark, C., Romano, M. 2019. Historical survey of kinematic and dynamic spacecraft simulators for laboratory experimentation of on-orbit proximity maneuvers. *Progress in Aerospace Sciences*. 110.

Appendix A Carrier cart design process

A.1 Quality functional deployment (House of qualities)

Figure A.1 shows the house of qualities which was constructed during the conceptual design of the carrier cart.



Priority	Customer Requirements / Engineering Characteristics	Cart size	Cart mass	Cart operation time	Max pressure	Translational acceleration	Rotational acceleration	Friction force	Cost per cart
	Improvement Direction	↓	↓	↑	↓	↑	↑	↓	↓
	Units	m^3	kg	min	Mpa	m/s^2	rad/s^2	N	R
5	Cart size < 40x40x40 cm	●	○	●	Δ	Δ	Δ	Δ	
5	Mass < 10 kg	○	●	Δ	Δ	○	○	○	
5	Operation time > 10 min	●	Δ	●	●	Δ	Δ		○
3	Cart interfaces various payloads	○	Δ			Δ	Δ		○
3	3 Degrees of freedom	○	○	Δ		○	○		●
3	Easy and convenient refill	○	○		Δ				Δ
4	Rigid & durable		○						○
2	Easily maintainable	○	○						Δ
4	Cost as low as possible			○	○				●
4	Safety towards operators	Δ	○		●	Δ	Δ		Δ
	Importance Ranking								
	Raw Score (Total = 684)	142	116	110	106	41	41	20	108
	Relative Weight [%]	20,76	16,959	16,082	15,497	5,9942	5,9942	2,924	15,789
	Rank Order	1	2	3	5	6	6	8	4
	Target Values	0,064	10	10	124	0,1	0,5	1e-3	35000

Legend
Symbols in Correlation Matrix
++ Strong Positive
\+ Positive
None
\- Negative
-- Strong Negative

Ratings in Relationship Matrix:
● - 9 (Strong)
○ - 3 (Medium)
Δ - 1 (Weak)

Figure A.1 - The quality functional deployment constructed for the carrier cart design process.

A.2 Concept sketches for the carrier cart

Figures A.2, A.3 and A.4 show three of the concepts developed for the carrier cart during the conceptual design phase.

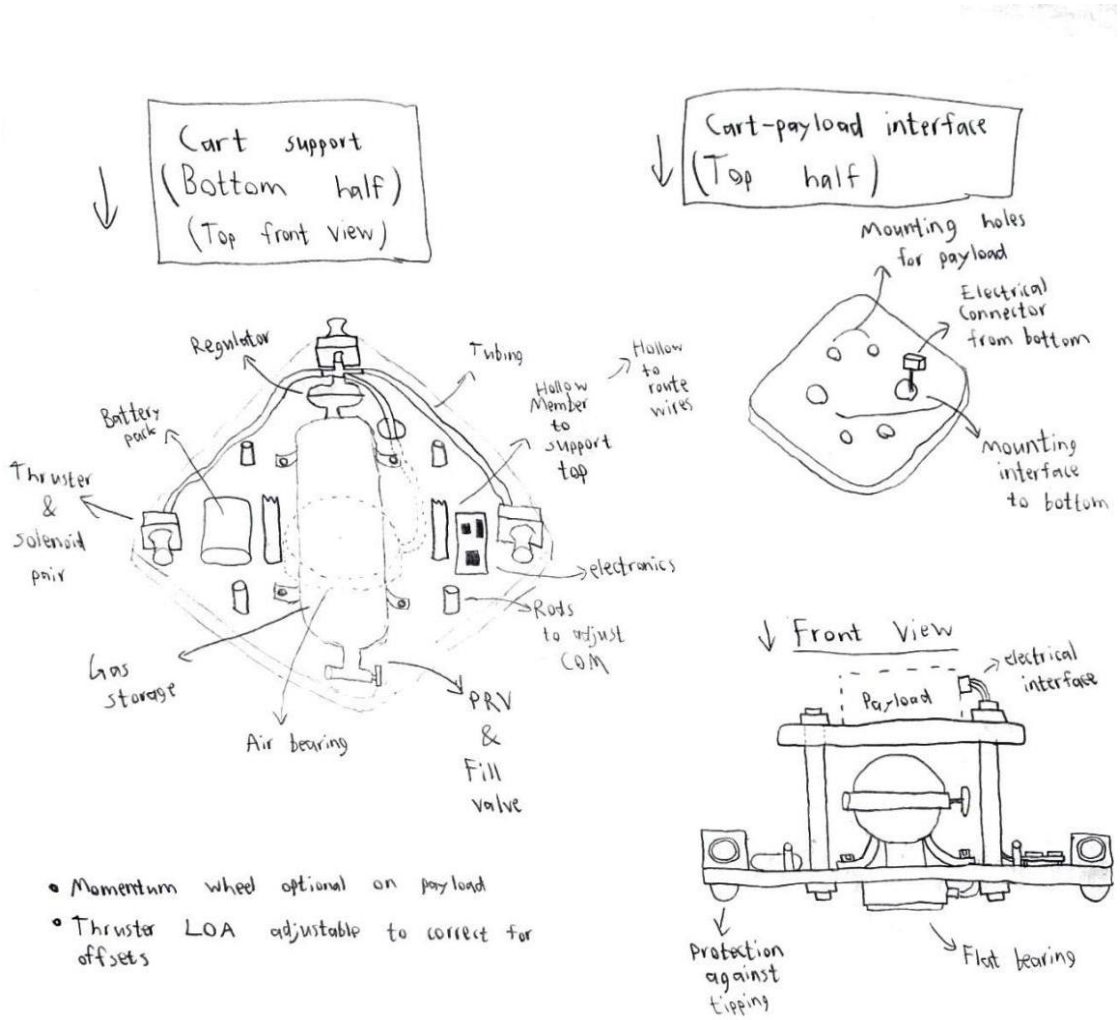


Figure A.2 - Concept 1 of the carrier cart conceptual design.

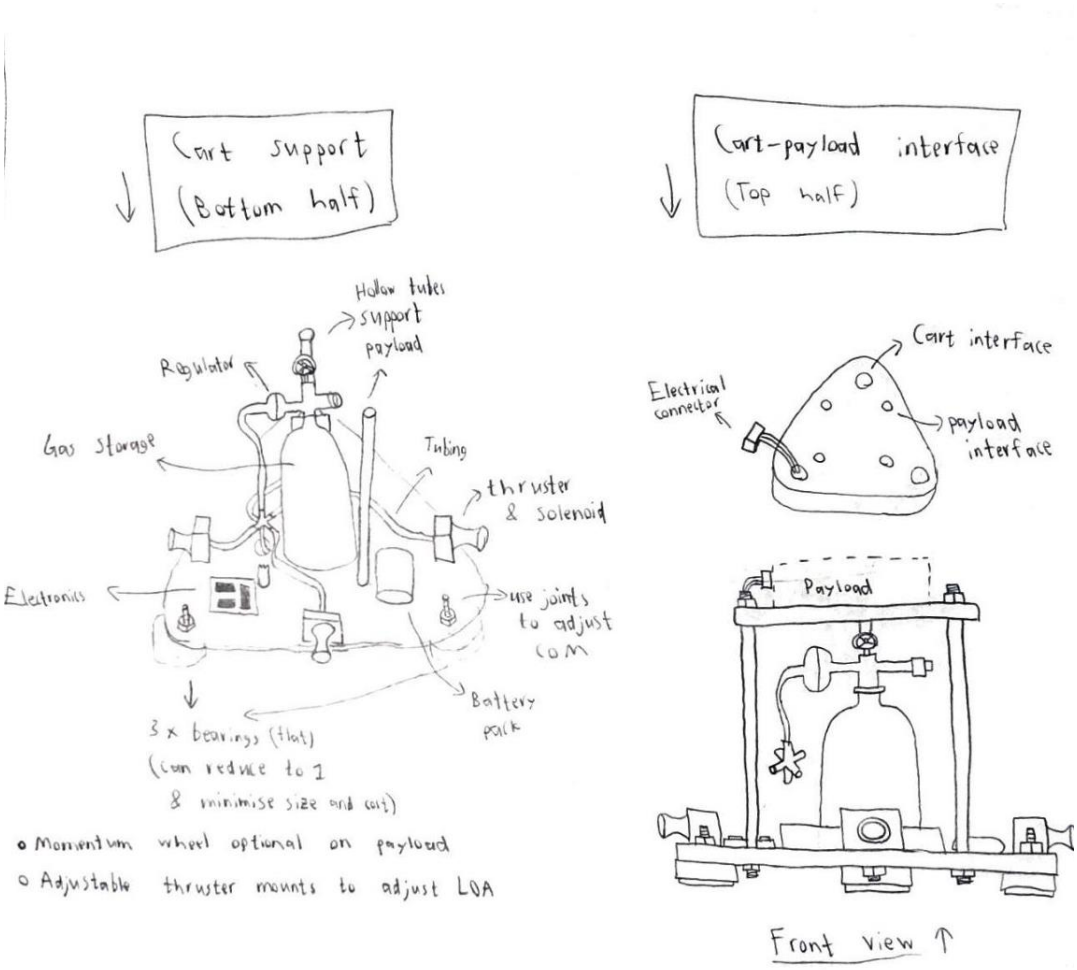


Figure A.3 - Concept 2 of the carrier cart conceptual design.

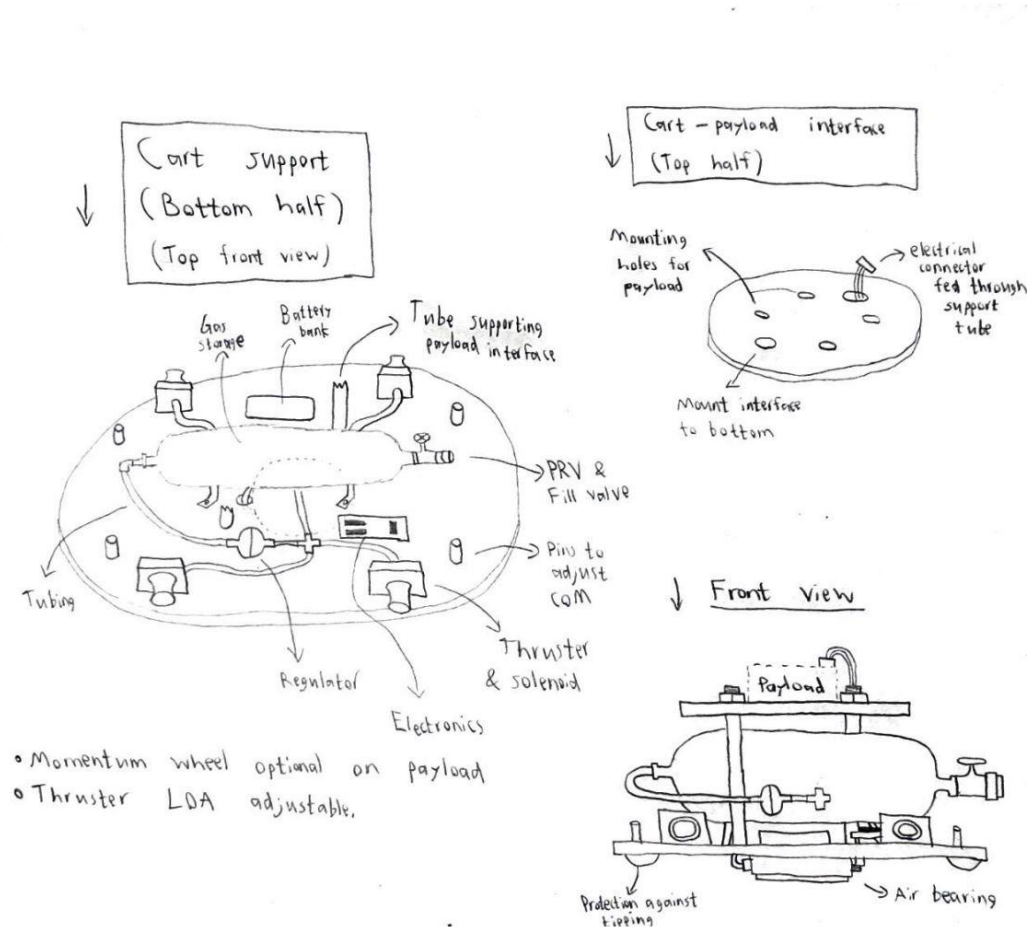


Figure A.4 - Concept 3 of the carrier cart conceptual design.

A.3 Bill of materials for the carrier cart pneumatic subsystem

Table A.1 shows the bill of materials required for a single carrier cart's pneumatic subsystem.

Table A.1 - The bill of materials for the pneumatic subsystem of the carrier cart.

Qty	Description	Supplier	Part number
1	DOUBLE-ENDED SAMPLE CYLINDER, 1 L	SWAGELOK	SS-4-MT
1	PRESSURE RELIEF VALVE	SWAGELOK	SS-4R3A5
1	SET SPRING KIT	SWAGELOK	117-R3A-K1-D
1	1 PIECE BALL VALVE	SWAGELOK	SS-43GF4
1	QUICK-CONNECT BODY	SWAGELOK	SS-QF4-B-4PM
1	PTFE-LINED, SS BRAIDED HOSE, 120 mm LENGTH	SWAGELOK	SS-4BHT-12
1	QUARTER-TURN PLUG VALVE	SWAGELOK	SS-4P4T4
1	PRESSURE GAUGE, 160 BAR, Ø63 mm	RS COMPONENTS	136-5201
1	PR REGULATOR, 248 BAR MAX INLET, 0 - 7 BAR OUT	SWAGELOK	KPR1FRA412A20000
1	PRESSURE GAUGE, 6 BAR, Ø50 mm	RS COMPONENTS	136-5168
7	FESTO AIR HOSE, 6 mm OD, 150 mm LENGTH	RS COMPONENTS	126-2777
1	FLAT ROUND AIR BEARING, Ø50 mm	NEW WAY AIR BEARINGS	S105001
4	MINIATURE SOLENOID VALVE, 2/2 NC	ASCO (VIA EMERSON)	411J1106HVS
2	¼NPT MALE TEE	SWAGELOK	SS-4-MT
2	¼NPT FEMALE ELBOW	SWAGELOK	SS-4-E
2	¼NPTM TO ¼BSPG F ADAPTOR	SWAGELOK	SS-4-AG-4
2	COPPER GASKET	SWAGELOK	CU-4-RG-2
2	¼NPTM TO SWAGELOK ¼TUBE FITTING	SWAGELOK	SS-400-1-4
1	¼NPT STREET TEE	SWAGELOK	SS-4-ST
1	¼NPTM TO ¼BSPR F ADAPTOR	SWAGELOK	SS-4-AT-4
1	¼BSPR M TO PUSH-IN TUBE, 6 mm OD TUBE	RS COMPONENTS	812-162
2	PUSH-IN TUBE CROSS, 6 mm OD TUBE	RS COMPONENTS	916-0928
5	PUSH-IN TUBE TO M5x0.8, 6 mm OD TUBE	RS COMPONENTS	265-5545

Appendix B Camera parameters

B.1 Camera parameters used for image streaming

The values assigned to camera parameters in the camera's driver software used during image streaming is tabulated in Table B.2.

Table B.2 - Values assigned to camera parameters used during image streaming.

Parameter name	Value assigned
Framerate	30 fps
Resolution	1440x1080
Gain	32
Exposure time	33333 µs
Brightness	0
Whitebalance red	70
Whitebalance blue	160
Whitebalance green	60

Appendix C Demonstration payload control simulation parameters

C.1 Low-level controller parameters used in the control system simulations

Table C.1 and Table C.2 list the parameters used for the low-level controllers during the simulations performed in Chapter 7.

Table C.3 - Values of the cart controller's low-level rotational controller used in the control simulations.

Control parameter	Final chosen value for practical tests
K_R	0.0452
T_D	3.01 s
α_D	0.0311

Table C.2 - Values of the cart controller's low-level heading velocity controller used in the control simulations.

Control parameter	Final chosen value for practical tests
K_T	6.39
T_I	12.5 s
α_I	10.0

Appendix D Demonstration payload practical test controller parameters

D.1 Tuned low-level controller parameters used in the demonstration control manoeuvre

Table D.1 and Table D.2 list the low-level controllers' parameters used for the practical implementation of the cart controller discussed in Chapter 8.

Table D.4 - Tuned values of the cart controller's low-level rotational controller used in the demonstration manoeuvre.

Control parameter	Final chosen value for practical tests
K_R	0.0226
T_D	3.01 s
α_D	0.0311

Table D.2 - Tuned values of the cart controller's low-level heading velocity controller used in the demonstration manoeuvre.

Control parameter	Final chosen value for practical tests
K_T	6.39
T_I	12.5 s
α_I	10.0